Concept 2.6
Bausteinbibliothek IEC
Teil: CONT_CTL

01/2007

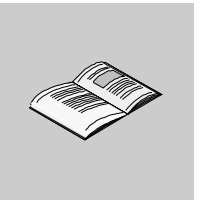
33002237.04

a brand of Schneider Electric

Telemecanique

www.telemecanique.com

Inhaltsverzeichnis



Sicherheitshinweise 17

Über dieses Buch 19

Teil I Allgemeines zur Bausteinbibliothek CONT_CTL 21
Übersicht 21

Kapitel 1 Parametrierung von Funktionen und Funktionsbausteinen 23
Parametrierung von Funktionen und Funktionsbausteinen 24

Kapitel 2 Allgemeine Informationen zur Baustein-Bibliothek CONT_CTL 27
Einleitung 27
Gruppen der Baustein-Bibliothek CONT_CTL 28
Betriebsarten 34
Abtasten 36
Fehlerverwaltung 37
Konvention 38

Teil II EFB-Beschreibungen (A bis PH) 39
Übersicht 39

Kapitel 3 ALIM: Anstiegsbegrenzer 2. Ordnung 41
Übersicht 41
Kurzbeschreibung 42
Darstellung 43
Detailbeschreibung 44
Laufzeitfehler 45

Kapitel 4 AUTOTUNE: Automatische Reglereinstellung 47
Übersicht 47
Kurzbeschreibung 48
Darstellung 49
Prinzip der automatischen Reglereinstellung 52
Identifikationsprinzip 54
Parametrierung 55
Reglerkopplung 58
Betriebsarten 59
Diagnose 60
Status der automatischen Reglereinstellung 61
Ursachen für einen Fehlstart 61
Ursachen für den Abbruch der automatischen Reglereinstellung 62
Generierung eines Test nach Beenden der automatischen Reglereinstellung 64
Laufzeitfehler 69

Kapitel 5 COMP_DB: Vergleich 71
Übersicht 71
Kurzbeschreibung 72
Darstellung 72
Detailbeschreibung 73
Laufzeitfehler 74

Kapitel 6 COMP_PID: Komplexer PID-Regler 75
Übersicht 75
Kurzbeschreibung 76
Darstellung 77
Strukturbild des komplexen PID-Reglers 80
Parametrierung des COMP_PID-Reglers 81
Antiwindup für COMP_PID 84
Auswahl des Reglertyps für COMP_PID 85
Stoßfreie Betriebsartenumschaltung 86
Auswahl der Betriebsart des COMP_PID 89
Detaillierte Formeln 92
Laufzeitfehler 94

Kapitel 7 DEADTIME: Totzeitglied 95
Übersicht 95
Kurzbeschreibung 96
Darstellung 97
Betriebsarten 98
Beispiel zum Verhalten des Funktionsbausteines 99
Laufzeitfehler 99

Kapitel 8 DELAY: Totzeitglied 101
Übersicht 101
Kurzbeschreibung 102
Darstellung 103
Betriebsarten 104
Beispiel zum Verhalten des Funktionsbausteines 105

Kapitel 9 DERIV: Differenzierer mit Glättung 107
Übersicht 107
Kurzbeschreibung 108
Darstellung 108
Formeln 109
Detailbeschreibung 110
Beispiel zum Funktionsbaustein 111
Laufzeitfehler 111

Kapitel 10 DTIME: Verzögerung 113
Übersicht 113
Kurzbeschreibung 114
Darstellung 115
Parametrierung 116
Initialisierungen und Betriebsarten 119
Beispiel zum Messen einer Durchflußmenge 120
Laufzeitfehler 121

Kapitel 11 FGEN: Funktionsgenerator 123
Übersicht 123
Kurzbeschreibung 124
Darstellung 125
Parametrierung 126
Funktionsauswahl 127
Funktionsdefinition 128
Diagramme der einzelnen Funktionen 131
Sonderfälle 135
Taktdiagramme 136

Kapitel 12 INTEG: Integrierer mit Begrenzung 139
Übersicht 139
Kurzbeschreibung 140
Darstellung 141
Detailbeschreibung 142
Laufzeitfehler 143

Kapitel 13 INTEGRATOR: Integrierer mit Begrenzung 145
Übersicht 145
Kurzbeschreibung 146
Darstellung 147
Detailbeschreibung 148
Laufzeitfehler 149

Kapitel 14 INTEGRATOR1: Integrierer mit Begrenzung 151
Übersicht 151
Kurzbeschreibung 152
Darstellung 153
Detailbeschreibung 154
Laufzeitfehler 155

Kapitel 15 K_SQRT: Quadratwurzel 157
Übersicht 157
Kurzbeschreibung 158
Darstellung 158
Laufzeitfehler 159

Kapitel 16 LAG: Verzögerungsglied 1. Ordnung 161
Übersicht 161
Kurzbeschreibung 162
Darstellung 163
Detailbeschreibung 164

Kapitel 17 LAG1: Verzögerungsglied 1. Ordnung 167
Übersicht 167
Kurzbeschreibung 168
Darstellung 169
Detailbeschreibung 170

Kapitel 18 LAG2: Verzögerungsglied 2. Ordnung 173
Übersicht 173
Kurzbeschreibung 174
Darstellung 175
Detailbeschreibung 176
Taktdiagramme 177

Kapitel 19 LAG_FILTER: Verzögerungsglied 1. Ordnung 179
Übersicht 179
Kurzbeschreibung 180
Darstellung 181
Detailbeschreibung 182

Kapitel 20 LDLG: PD-Glied mit Glättung 183
Übersicht 183
Kurzbeschreibung 184
Darstellung 185
Detailbeschreibung 186
Beispiele zum Funktionsbaustein LDLG 187

Kapitel 21 LEAD: Differenzierer mit Glättung 189
Übersicht 189
Kurzbeschreibung 190
Darstellung 191
Detailbeschreibung 192

Kapitel 22 LEAD_LAG: PD-Glied mit Glättung 193
Übersicht 193
Kurzbeschreibung 194
Darstellung 195
Detailbeschreibung 196
Beispiele zum Funktionsbaustein LEAD_LAG 197
Laufzeitfehler 199

Kapitel 23 LEAD_LAG1: PD-Glied mit Glättung 201
Übersicht 201
Kurzbeschreibung 202
Darstellung 203
Detailbeschreibung 204
Beispiele zum Funktionsbaustein LEAD_LAG1 205

Kapitel 24 LIMV: Anstiegsbegrenzer 1. Ordnung 207
Übersicht 207
Kurzbeschreibung 208
Darstellung 209
Detailbeschreibung 210
Laufzeitfehler 211

Kapitel 25 MFLOW: Baustein für Massendurchfluß 213
Übersicht 213
Kurzbeschreibung 214
Darstellung 215
Detailbeschreibung 216
Laufzeitfehler 217

Kapitel 26 MS: Handsteuerung eines Ausgangs 219
Übersicht 219
Kurzbeschreibung 220
Darstellung 221
Detailbeschreibung 223
Beispiel 227
Laufzeitfehler 228

Kapitel 27 MULDIV_W: Multiplikation/Division 231
Übersicht 231
Kurzbeschreibung 232
Darstellung 232
Laufzeitfehler 233

Kapitel 28 PCON2: Zweipunkt-Regler 235
Übersicht 235
Kurzbeschreibung 236
Darstellung 237
Detailbeschreibung 238
Laufzeitfehler 240

Kapitel 29 PCON3: Dreipunkt-Regler 241
Übersicht 241
Kurzbeschreibung 242
Darstellung 243
Detailbeschreibung 245
Laufzeitfehler 247

Kapitel 30 PD_or_PI: Strukturumschaltung PD- /PI-Regler 249
Übersicht 249
Kurzbeschreibung 250
Darstellung 251
Strukturbild des Funktionsbausteins PD_or_PI 253
Detailbeschreibung 254
Detailierte Formeln 257
Laufzeitfehler 259

Kapitel 31 PDM: Pulsdauermodulation 261
Übersicht 261
Kurzbeschreibung 262
Darstellung 263
Detailbeschreibung 264
Laufzeitfehler 268

Teil III EFB-Beschreibungen (PI bis Z). 269
Übersicht 269

Kapitel 32 PI: PI-Regler 271
Übersicht 271
Kurzbeschreibung 272
Darstellung 273
Formeln 275
Parametrierung 276
Betriebsarten 278
Beispiel zum PI-Regler 279
Laufzeitfehler 280

Kapitel 33 PI1: PI-Regler 281
Übersicht 281
Kurzbeschreibung 282
Darstellung 283
Formeln 284
Parametrierung 285
Betriebsarten 287
Beispiel zum PI1-Regler 288
Laufzeitfehler 289

Kapitel 34 PI_B: Einfacher PI-Regler 291
Übersicht 291
Kurzbeschreibung 292
Darstellung 293
Formeln 295
Parametrierung 296
Detaillierte Gleichungen 300
Laufzeitfehler 302

Kapitel 35 PID: PID-Regler 305
Übersicht 305
Kurzbeschreibung 306
Darstellung 307
Strukturbild des Funktionsbausteins PID 309
Parametrierung des PID-Reglers 310
Betriebsarten 312
Detaillierte Formeln 315
Laufzeitfehler 317

Kapitel 36 PID1: PID-Regler 319
Übersicht 319
Kurzbeschreibung 320
Darstellung 321
Strukturbild des Funktionsbausteins PID1 323
Parametrierung des PID1-Reglers 324
Betriebsarten 326
Detaillierte Formeln 329
Laufzeitfehler 330

Kapitel 37 PID_P: PID-Regler mit paralleler Struktur 331
Übersicht 331
Kurzbeschreibung 332
Darstellung 333
Parametrierung des PID_P-Reglers 335
Betriebsarten 337
Detaillierte Formeln 338
Laufzeitfehler 339

Kapitel 38 PID_PF: PID-Regler mit paralleler Struktur 341
Übersicht 341
Kurzbeschreibung 342
Darstellung 343
Parametrierung des PID_PF-Reglers 345
Betriebsarten 347
Detaillierte Formeln 348
Laufzeitfehler 349

Kapitel 39 PIDFF: Kompletter PID-Regler 351
Übersicht 351
Kurzbeschreibung 352
Darstellung 353
Formeln 355
Strukturbild des PIDFF-Reglers 357
Parametrierung 358
Betriebsarten 363
Detaillierte Gleichungen 364
Detaillierte Gleichungen: Inkrementaler Algorithmus PID-Regler 367
Detaillierte Gleichungen: Inkrementaler Algorithmus Integral-Betrieb 369
Beispiele zum Baustein PIDFF 371
Laufzeitfehler 376

Kapitel 40 PIDP1: PID-Regler mit paralleler Struktur 379
Übersicht 379
Kurzbeschreibung 380
Darstellung 381
Parametrierung des PIDP1-Reglers 383
Betriebsarten 385
Detaillierte Formeln 386
Laufzeitfehler 388

Kapitel 41 PIP: PIP-Kaskaden-Regler 389
Übersicht 389
Kurzbeschreibung 390
Darstellung 391
Strukturbild des Funktionsbausteins PIP 393
Parametrierung des PIP-Kaskaden-Reglers 394
Betriebsarten 396
Detaillierte Formeln 398
Laufzeitfehler 399

Kapitel 42 PPI: PPI-Kaskaden-Regler 401
Übersicht 401
Kurzbeschreibung 402
Darstellung 403
Strukturbild des Funktionsbausteins PPI 405
Parametrierung des PPI-Kaskaden-Reglers 406
Betriebsarten 408
Detaillierte Formeln 409
Laufzeitfehler 410

Kapitel 43 PWM: Pulsbreitenmodulation 411
Übersicht 411
Kurzbeschreibung 412
Darstellung 413
Formeln 414
Detailbeschreibung 415
Beispiele zum Baustein PWM 418

Kapitel 44 PWM1: Pulsbreitenmodulation 421
Übersicht 421
Kurzbeschreibung 422
Darstellung 423
Formeln 424
Detailbeschreibung 425
Beispiel zum Baustein PWM1 427

Kapitel 45 QDTIME: Totzeitglied 429
Übersicht 429
Kurzbeschreibung 430
Darstellung 431
Detailbeschreibung 432

Kapitel 46 QPWM: Pulsbreitenmodulation (einfach) 435
Übersicht 435
Kurzbeschreibung 436
Darstellung 437
Formeln 438
Detailbeschreibung 439
Beispiel zum Baustein QPWM 441

Kapitel 47 RAMP: Rampengenerator 443
Übersicht 443
Kurzbeschreibung 444
Darstellung 445
Detailbeschreibung 446
Laufzeitfehler 448

Kapitel 48 RATIO: Verhältnisregler 449
Übersicht 449
Kurzbeschreibung 450
Darstellung 451
Detailbeschreibung 452
Laufzeitfehler 454

Kapitel 49 SCALING: Skalierung 455
Übersicht 455
Kurzbeschreibung 456
Darstellung 456
Parametrierung 457
Laufzeitfehler 458

Kapitel 50 SCON3: Dreipunkt-Schrittregler 459
Übersicht 459
Kurzbeschreibung 460
Darstellung 461
Detailbeschreibung 462
Laufzeitfehler 464

Kapitel 51 SERVO: Steuerung für elektrische Servomotoren 465
Übersicht 465
Kurzbeschreibung 466
Darstellung 466
Parametrierung 468
Algorithmen zum Funktionsbaustein SERVO 470
Betriebsarten 471
Beispiele zum Funktionsbaustein SERVO 472
Laufzeitfehler 479

Kapitel 52 SMOOTH_RATE: Differenzierer mit Glättung 481
Übersicht 481
Kurzbeschreibung 482
Darstellung 482
Formeln zum Funktionsbaustein SMOOTH_RATE 483
Detailbeschreibung 484

Kapitel 53 SP_SEL: Sollwertschalter 487
Übersicht 487
Kurzbeschreibung 488
Darstellung 489
Detailbeschreibung 491
Laufzeitfehler 494

Kapitel 54 SPLRG: Steuerung von 2 Stellgliedern 495
Übersicht 495
Kurzbeschreibung 496
Darstellung 497
Detailbeschreibung 498
Laufzeitfehler 500

Kapitel 55 STEP2: Zweipunktregler 501
Übersicht 501
Kurzbeschreibung 502
Darstellung 503
Detailbeschreibung 504
Laufzeitfehler 506

Kapitel 56 STEP3: Dreipunktregler 507
Übersicht 507
Kurzbeschreibung 508
Darstellung 509
Detailbeschreibung 510
Laufzeitfehler 512

Kapitel 57 SUM_W: Summierer 513
Übersicht 513
Kurzbeschreibung 514
Darstellung 514
Laufzeitfehler 514

Kapitel 58 THREEPOINT_CON1: Dreipunkt-Regler 515
Übersicht 515
Kurzbeschreibung 516
Darstellung 516
Detailbeschreibung 518
Laufzeitfehler 521

Kapitel 59 THREE_STEP_CON1: Dreipunkt-Schrittregler 523
Übersicht 523
Kurzbeschreibung 524
Darstellung 525
Detailbeschreibung 526
Laufzeitfehler 528

Kapitel 60 TOTALIZER: Integrierer 529
Übersicht 529
Kurzbeschreibung 530
Darstellung 531
Formeln 533
Detailbeschreibung 534
Laufzeitfehler 538

Kapitel 61 TWOPOINT_CON1: Zweipunkt-Regler 539
Übersicht 539
Kurzbeschreibung 540
Darstellung 541
Detailbeschreibung 542
Laufzeitfehler 544

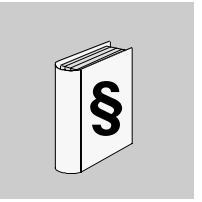
Kapitel 62 VEL_LIM: Anstiegsbegrenzer 545
Übersicht 545
Kurzbeschreibung 546
Darstellung 547
Detailbeschreibung 548
Laufzeitfehler 549

Kapitel 63 VLIM: Anstiegsbegrenzer 1. Ordnung 551
Übersicht 551
Kurzbeschreibung 552
Darstellung 553
Detailbeschreibung 554
Laufzeitfehler 555

Glossar 557

Index 583

Sicherheitshinweise



Wichtige Informationen

HINWEIS

Lesen Sie diese Anweisungen gründlich durch und machen Sie sich mit dem Gerät vertraut, bevor Sie es installieren, in Betrieb nehmen oder warten. Die folgenden Hinweise können an verschiedenen Stellen in dieser Dokumentation enthalten oder auf dem Gerät zu lesen sein. Die Hinweise warnen vor möglichen Gefahren oder machen auf Informationen aufmerksam, die Vorgänge erläutern bzw. vereinfachen.

Erscheint dieses Symbol zusätzlich zu einem Warnaufkleber, bedeutet dies, dass die Gefahr eines elektrischen Schlags besteht und die Nichtbeachtung des Hinweises Verletzungen zur Folge haben kann.

Dies ist ein allgemeines Warnsymbol. Es macht Sie auf mögliche Verletzungsgefahren aufmerksam. Beachten Sie alle unter diesem Symbol aufgeführten Hinw um Verletzungen oder Unfälle mit Todesfälle zu vermeiden.

⚠ GEFAHR

GEFAHR macht auf eine unmittelbar gefährliche Situation aufmerksam, die bei Nichtbeachtung **unweigerlich** einen schweren oder tödlichen Unfall zur Folge hat.

⚠ WARNUNG

WARNUNG macht auf eine möglicherweise gefährliche Situation aufmerksam, die bei Nichtbeachtung **unter Umständen** einen schweren oder tödlichen Unfall oder Beschädigungen an Geräten zur Folge haben kann.

⚠ VORSICHT

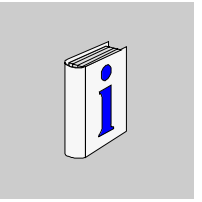
VORSICHT macht auf eine möglicherweise gefährliche Situation aufmerksam, die bei Nichtbeachtung **unter Umständen** einen schweren oder tödlichen Unfall oder Beschädigungen an Geräten zur Folge hat.

BITTE BEACHTEN

Elektrische Geräte dürfen nur von Fachpersonal installiert, betrieben, gewartet und instand gesetzt werden. Schneider Electric haftet nicht für Schäden, die aufgrund der Verwendung dieses Materials entstehen.

© 2007 Schneider Electric. Alle Rechte vorbehalten.

Über dieses Buch



Auf einen Blick

Ziel dieses Dokuments

Diese Dokumentation soll Ihnen bei der Konfiguration der Funktionen und Funktionsbausteine helfen.

Gültigkeitsbereich

Diese Dokumentation gilt für Concept 2.6 unter Microsoft Windows 98, Microsoft Windows 2000, Microsoft Windows XP und Microsoft Windows NT 4.x.

Hinweis: Weitere aktuelle Hinweise finden Sie in der Datei README von Concept.

Weiterführende Dokumentation

Titel	Referenz-Nummer
Concept Installationsanleitung	840 USE 502 02
Concept Benutzerhandbuch	840 USE 503 02
Concept-EFB User Manual	840 USE 505 00
Concept LL984 Bausteinbibliothek	840 USE 506 02

Sie können diese technischen Veröffentlichungen sowie andere technische Informationen von unserer Website herunterladen: *www.telemecanique.com*

Benutzerkommentar

Ihre Anmerkungen und Hinweise sind uns jederzeit willkommen. Senden Sie sie einfach an unsere E-mail-Adresse: techpub@schneider-electric.com

Allgemeines zur Bausteinbibliothek CONT_CTL

I

Übersicht

Einleitung

Dieser Abschnitt enthält allgemeine Informationen zur Bausteinbibliothek CONT_CTL.

Inhalt dieses Teils

Dieser Teil enthält die folgenden Kapitel:

Kapitel	Kapitelname	Seite
1	Parametrierung von Funktionen und Funktionsbausteinen	23
2	Allgemeine Informationen zur Baustein-Bibliothek CONT_CTL	27

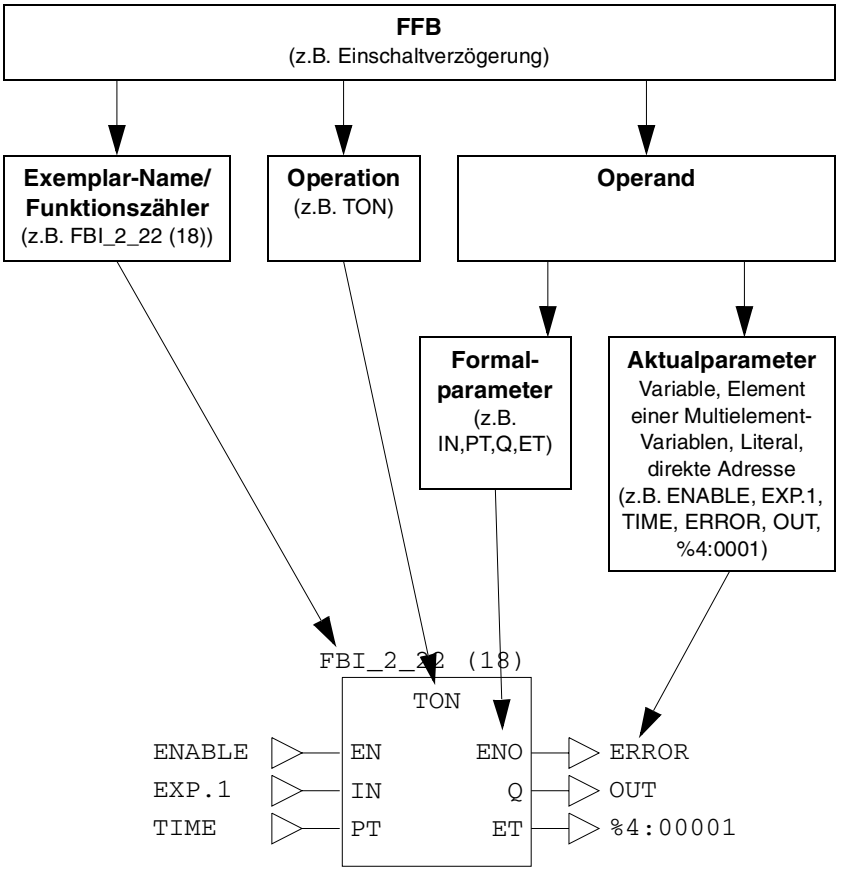
Parametrierung von Funktionen und Funktionsbausteinen

1

Parametrierung von Funktionen und Funktionsbausteinen

Allgemeines

Jeder FFB setzt sich zusammen aus einer Operation, den zur Operation notwendigen Operanden und einem Exemplar-Name/Funktionszähler.



Operation

Die Operation bestimmt, welche Funktionalität durch den FFB ausgeführt werden soll, z.B. Schieberegister, Wandeloperationen.

Operand

Der Operand legt fest, womit die Operation ausgeführt werden soll. Er besteht bei FFBs aus Formalparameter und Aktualparameter.

Formalparameter/ Aktualparameter

Der Formalparameter ist ein Platzhalter für einen Operanden. Bei der Parametrierung wird dem Formalparameter ein aktueller Parameter zugewiesen (Aktualparameter).

Der Aktualparameter kann eine Variable, eine Multielement-Variable, ein Element einer Multielement-Variablen, ein Literal oder eine direkte Adresse sein.

Bedingter/ Unbedingter Aufruf

Jeder FFB verfügt über die Möglichkeit des "bedingten" oder "unbedingten" Aufrufs. Die Bedingung wird über eine Vorverknüpfung des Eingangs EN realisiert.

- EN eingeblendet
  bedingter Aufruf (nur wenn EN = 1, wird der FFB bearbeitet)
- EN ausgeblendet
  unbedingter Aufruf (FFB wird immer bearbeitet)

Hinweis: Wird der EN-Eingang nicht parametriert, muß er ausgeblendet werden. Da nicht parametrierte Eingänge automatisch mit einer "0" belegt werden, würde der FFB sonst nie bearbeitet.

Hinweis: Bei gesperrten Funktionsbausteinen (EN = 0) mit interner Zeitfunktion (z.B. DELAY) läuft die Zeit scheinbar weiter, da sie mit Hilfe der Systemuhr berechnet wird und damit unabhängig vom Programmzyklus und der Freigabe des Bausteins ist.

Aufruf von Funktionen und Funktionsbausteinen in IL und ST

Den Aufruf von Funktionen und Funktionsbausteinen in IL (Anweisungsliste) und ST (Strukturierter Text) entnehmen Sie bitte den enstprechenden Kapiteln des Benutzerhandbuches.

Allgemeine Informationen zur Baustein-Bibliothek CONT_CTL

2

Einleitung

Übersicht

Dieser Abschnitt enthält allgemeine Informationen über die Baustein-Bibliothek CONT_CTL.

Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Gruppen der Baustein-Bibliothek CONT_CTL	28
Betriebsarten	34
Abtasten	36
Fehlerverwaltung	37
Konvention	38

Gruppen der Baustein-Bibliothek CONT_CTL

Übersicht über die Gruppen

Die "Continuous Control"(CONT-CTL)-Bibliothek besteht aus 7 Gruppen mit Elementaren Funktionsbausteinen (EFBs):

Gruppe	Inhalt
CLC	Enthält regelungstechnische Funktionsbausteine wie z.B. Filter, Regler, Integratoren und Totzeitglieder
CLC_PRO	Enthält eine weitere Auswahl von regelungstechnischen Funktionsbausteinen
Conditioning	EFBs zur Verarbeitung des Meßwertes oder einer anderen diskreten Größe
Controller	Regler- EFBs und automatische Reglereinstellungs-Bausteine
Mathematics	EFBs für Mathematische Regelungsfunktionen
Output Processing	EFBs zur Steuerung der verschiedenen Stellgliedertypen
Setpoint Management	EFBS zur Generierung und Auswahl des Sollwertes

Gruppe "CLC"

Diese Gruppe enthält die folgenden EFBs:

Baustein	Bedeutung
DELAY	Totzeitglied
INTEGRATOR1	Integrierer mit Begrenzung (Betriebsarten Hand, Halt, Automatik)
LAG1	Verzögerungsglied 1. Ordnung
LEAD_LAG1	PD-Glied mit Glättung
LIMV	Anstiegsbegrenzer 1.Ordnung
PI1	PI-Regler
PID1	PID-Regler
PIDP1	PID-Regler mit paralleler Struktur
SMOOTH_RATE	Differenzierer mit Glättung
THREEPOINT_CON1	Dreipunkt-Regler
THREE_STEP_CON1	Dreipunkt-Schrittregler
TWOPOINT_CON1	Zweipunktregler

Gruppe "CLC_PRO"

Diese Gruppe enthält die folgenden EFBs:

Baustein	Bedeutung
ALIM	Anstiegsbegrenzer 2. Ordnung
COMP_PID	Komplexer PID-Regler
DEADTIME	Totzeitglied
DERIV	Differenzierer mit Glättung
FGEN	Funktionsgenerator
INTEG	Integrierer mit Begrenzung
LAG	Verzögerungsglied 1. Ordnung
LAG2	Verzögerungsglied 2. Ordnung
LEAD_LAG	PD-Glied mit Glättung
PCON2	Zweipunkt-Regler
PCON3	Dreipunkt-Regler
PD_or_PI	Strukturumschaltung PD-/PI-Regler
PDM	Pulsdauermodulation
PI	PI-Regler
PID	PID-Regler
PID_P	PID-Regler mit paralleler Struktur
PIP	PIP-Kaskadenregler
PPI	PPI-Kaskadenregler
PWM	Pulsbreitenmodulation
QPWM	Pulsbreitenmodulation (einfach)
SCON3	Dreipunkt-Schrittregler
VLIM	Anstiegsbegrenzer 1. Ordnung

Gruppe "Conditioning"

Diese Gruppe enthält die EFBs für Verarbeitungsvorgänge, die den Reglern im allgemeinen vorgeschaltet sind wie die Verarbeitung des Meßwertes der Regelgröße, der Störgrößen oder anderer diskreter Größen.

Über Filter und andere klassische Funktionen hinaus beinhaltet diese Gruppe ebenfalls Verzögerungs- und Summationsfunktionen.

Diese Gruppe enthält die folgenden EFBs:

Baustein	Bedeutung
DTIME	Verzögerungsfunktion, für eine erhöhte Präzision oder zur dynamischen (Online-) Änderung des Verzögerungswertes
INTEGRATOR	Integrierer mit Begrenzung (Betriebsarten Tracking und Automatik)
LAG_FILTER	Verzögerungsglied 1. Ordnung
LDLG	PD-Glied mit Glättung (Phalsenvorschub/-verzögerung)
LEAD	Differenzierer mit Glättung
MFLOW	Regler für Massendurchfluß, z.B. zur Verarbeitung der Differentialdruckmessung eines Drosselgerätes
QDTIME	Totzeitglied, Verzögerungsfunktion zur schnellen Parametrierung (Q = Quick)
SCALING	Skalierung aller diskreten Größen
TOTALIZER	Integrierer, um einen Durchfluß zu integrieren und somit ein Durchflußvolumen zu berechnen. Mit diesem EFB können sehr kleine Werte berücksichtigt werden, auch dann, wenn das Gesamtvolumen groß ist. Er besitzt einen Teilmengen- und einen Gesamtmengenzähler.
VEL_LIM	Anstiegsbegrenzung eines Eingangs oder einer Zwischengröße

Gruppe "Controller"

Diese Gruppe beinhaltet u.a. einen Baustein zur automatischen Reglereinstellung (AUTOTUNE). Dieser Baustein ist auf die Reglerbausteine PI_B und PIDFF abgestimmt. Dadurch können selbsteinstellende Reglerapplikationen programiert werden.

Diese Gruppe enthält die folgenden EFBs:

Baustein	Bedeutung
AUTOTUNE	Automatische Reglereinstellung
PI_B	Einfacher PI-Regler
PIDFF	Kompletter PID-Regler
STEP2	Zweipunktregler
STEP3	Dreipunktregler

Gruppe "Mathematics"

Im Bereich der Regelung werden arithmetische Funktionen häufig im Zusammenhang mit Unempfindlichkeitszonen und Gewichtungen verwendet.

Ausgehend von diesem Prinzip umfaßt diese Gruppe direkt verwendbare arithmetische Funktionen:

- Multiplikation / Division mit Gewichtung: MULDIV_W
- Summation mit Gewichtung: SUM_W
- Vergleich mit Unempfindlichkeitszone und Hysterese: COMP_DB
- Quadratwurzel mit Trennpunkt und Gewichtung: K_SQRT

Diese Gruppe enthält die folgenden EFBs:

Baustein	Bedeutung
COMP_DB	Vergleich
K_SQRT	Quadratwurzel
MULDIV_W	Multiplikation / Division
SUM_W	Summierer

Gruppe "Output Processing"

Häufig ist es nicht möglich, den Reglerausgang direkt zu verwenden, um das Stellglied zu steuern.

Wenn beispielsweise, wie dies bei vielen Prozessen der Fall ist, elektrische Servomotoren zum Einsatz kommen, muß dem Regler ein SERVO-Funktionsbaustein nachgeschaltet werden.

Wenn zwei Stellglieder auf dieselbe Größe einwirken, ist der Funktionsbaustein SPLRG zu verwenden. Dieser Funktionsbaustein funktioniert sowohl als Dreipunkt-Schrittregler (bei entgegengesetzter Wirkung der Stellglieder), wie auch in der Betriebsart "Split-range" (bei gleichgerichteter Wirkung der Stellglieder).

Der Baustein PWM1 ermöglicht eine Pulsweitenmodulation von z.B. einer Stellgröße eines vorgeschalteten kontinuierlichen Reglers (PI, PID).

Obwohl alle Reglerbausteine in der Betriebsart Hand arbeiten können, ist es häufig erforderlich, hierfür den Funktionsbaustein MS zu verwenden.

Dieser Baustein erlaubt eine erweiterte Steuerung der Betriebsart Hand

- Die zu steuernde Variable ist nicht direkt der Reglerausgang
- Der Ausgang wird nicht über einen Regelkreis gesteuert
- Der Regelkreis hat ein langes Abtastintervall (1 s und mehr)

Diese Gruppe enthält die folgenden EFBs:

Baustein	**Bedeutung**
MS	Handsteuerung eines Ausgangs
PWM1	Pulsbreitenmodulation
SERVO	Steuerung für elektrische Servomotoren
SPLRG	Steuerung von 2 Stellgliedern

Gruppe "Setpoint Management"

Die klassische Funktion 'Auswahl des Sollwertes' ist nicht in den Reglerbausteinen integriert, sondern in dem Funktionsbaustein SP_SEL. Diese modulare Struktur ermöglicht eine größere Flexibilität und verbesserten Bedienkomfort, ohne dabei auf erweiterte Funktionen zu verzichten.

Hierzu zählen:

- Die Nachführung des Istwertes, wenn der Regelkreis auf Handbetrieb eingestellt ist
- Die stoßfreie Umschaltung intern/extern
- Die stoßfreie Umschaltung extern/intern (mit Nachführung des Sollwertes)

Zwei andere Funktionsbausteine ermöglichen die Generierung des dem Regler aufzuschaltenden Sollwertes: der Funktionsbaustein RATIO, der zur Steuerung einer Größe in Abhängigkeit von einer anderen Größe verwendet wird (Verhältnisregelung) und der Baustein RAMP, der die Generierung eines Sollwertes in Rampenform ermöglicht.

Diese Gruppe enthält die folgenden EFBs:

Baustein	**Bedeutung**
RAMP	Rampengenerator
RATIO	Verhältnisregler
SP_SEL	Sollwertschalter

Betriebsarten

Betriebsarten

Verschiedene Funktionsbausteine verfügen über eine integrierte Betriebsartensteuerung.

Es kann zwischen folgenden Betriebsarten gewählt werden:
- Tracking (Nachführung)
- Hand/Automatik.

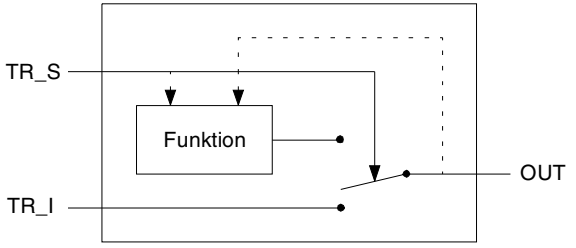
Weiterhin wird die Prioritätsreihenfolge der Betriebsarten erläutert.

Tracking

Diese Betriebsart ermöglicht es, einen Funktionsbaustein auf die Betriebsart 'Folgeregler' einzustellen. Zwei Eingänge ermöglichen die Steuerung dieser Betriebsart: ein Binäreingang TR_S (TRacking Switch), sowie ein Signaleingang TR_I (TRacking Input). Wenn sich ein Funktionsbaustein im Trackingbetrieb befindet (TR_S = 1), wird seinem Hauptausgang (z.B. OUT bei einem PIDFF-Regler) der Eingangswert TR_I aufgeschaltet und die internen Variablen der verschiedenen Algorithmen aktualisiert. So wird ein stoßfreies Umschalten gewährleistet, wenn der Funktionsbaustein auf Hand- oder Automatikbetrieb umgestellt wird.

Im Trackingbetrieb wird der Ausgang OUT des FFBs durch den Eignang TR_I gesteuert.

Betriebsart Tracking



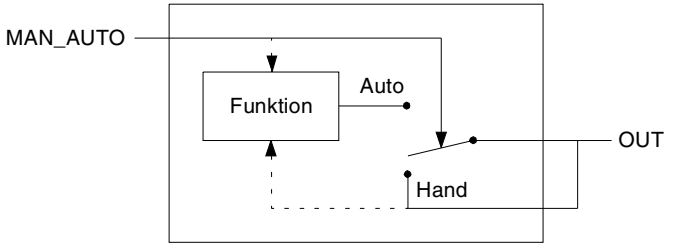
Diese Betriebsart kann in verschiedenen Fällen eingesetzt werden:
- Initialisierung bei der Startphase,
- Betriebsart Tracking bei redundanter SPS, um einen stoßfreien Start des Standby-Gerätes zu gewährleisten,
- Steuerung der Betriebsart über ein Programm, um beispielsweise die direkte Steuerung der Stellgröße zu vermeiden, wenn gerade eine automatische Reglereinstellung abläuft etc.

Dem Ausgang des Funktionsbausteins kann eine Begrenzung aufgeschaltet werden, wenn sich dieser in der Betriebsart Tracking befindet: dies ist für die einzelnen Funktionsbausteine individuell zu entscheiden.

Hand/Automatik

Wenn ein Funktionsbaustein sich in der Betriebsart Automatik befindet, errechnet sein Algorithmus den auf den Ausgang aufzuschaltenden Wert. Über die Betriebsart Hand kann die Einstellung des Hauptausgangs (OUT) eines Funktionsbausteins blockiert werden, um dann z.B. über Bedienerdialog gesteuert zu werden. Der Eingang MAN_AUTO ermöglicht die Steuerung dieser Betriebsart (0 : Hand, 1 : Automatik).

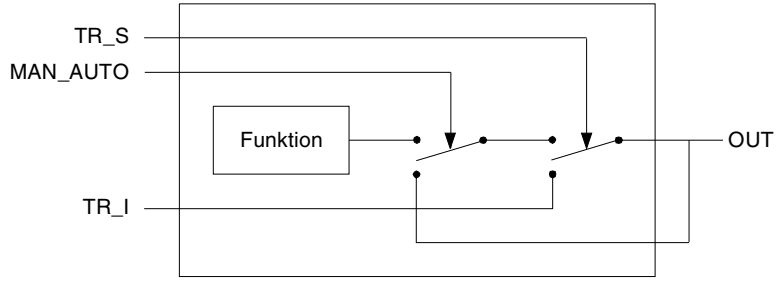
Betriebsart Hand/Automatik



Der Funktionsbaustein liest dennoch diesen Ausgang und ermöglicht so ein stoßfreies Umschalten zwischen den Betriebsarten Hand <-> Automatik. Dem Ausgang des Funktionsbausteins kann eine Begrenzung aufgeschaltet werden, wenn sich dieser in der Betriebsart Hand oder Automatik befindet: dies ist individuell für jeden Funktionsbaustein zu entscheiden.

Prioritätenreihenfolge der Betriebsarten

Wenn ein Funktionsbaustein über beide Betriebsarten verfügt, hat die Betriebsart Tracking Vorrang vor der Betriebsart Hand/Automatik:



Um eine bessere Übersicht zu gewährleisten, sind die Verbindungen zwischen der Funktion und der Betriebsart des Funktionsbausteins nicht dargestellt. Gleiches gilt für den effektiv aufgeschalteten Stellwert.

Abtasten

Abtasten

Die Regelalgorithmen basieren auf Abtastwerten, bei denen das Zeitintervall zwischen zwei aufeinanderfolgenden Zyklen zu berücksichtigen ist. Die Funktionsbausteine berechnen automatisch den Wert dieses Intervalls, d.h. sie können in den Concept-Sections an einer beliebigen Stelle gesetzt werden, ohne das Zeitmanagement berücksichtigen zu müssen.

Folgende Regelfunktionen können mit festem Zeitintervall durchgeführt werden :

- Laufzeit-Optimierung des SPS-Programms durch Aufteilen der Regeloperationen auf mehrere Zyklen,
- verbesserte Regelqualität, indem ein zu häufiges Abtasten der Regelkreise verhindert wird
- Minimierung der Beanspruchung der Stellglieder

Hierzu kann beispielsweise der Funktionsbaustein SAMPLETM verwendet werden, der an den Eingang EN der abzutastenden Funktionsbausteine anzuschließen ist.

Wenn das Abtastintervall der Regelkreise 1 Sekunde übersteigt, ist der Funktionsbaustein *MS: Handsteuerung eines Ausgangs, S. 219* den Regler-Funktionsbausteinen *PIDFF: Kompletter PID-Regler, S. 351* und *PI_B: Einfacher PI-Regler, S. 291* nachzuschalten, so daß die Regelkreise unabhängig von den Abtastintervallen manuell gesteuert werden können.

Fehlerverwaltung

Prinzip

Die meisten Funktionsbausteine der Gruppen "Conditioning", "Controller", "Output Processing" und "Setpoint Management" verfügen über ein STATUS-Ausgangswort. Die von diesen Funktionsbausteinen verwendeten Fehlererfassungs- und Melderoutinen werden in diesem Kapitel beschrieben.

Jedes Bit des STATUS-Parameters kann zur Meldung eines Fehlers, eines Alarms oder einer Information verwendet werden. Die Bedeutung der ersten 8 Bits des STATUS-Wortes ist bei allen Baugruppen dieselbe. Die Bedeutung der Folgebits (Bits 8 bis 15) ist bei jedem Funktionsbaustein verschieden.

Statuswort

Folgende Tabelle stellt die Bedeutung der allen Funktionsbausteinen gemeinsamen Bits im ersten Byte des STATUS-Wortes dar. Weitere Informationen finden Sie in der Beschreibung des jeweiligen Funktionsbausteins.

Bit	Bedeutung	Typ
Bit 0 = 1	Fehler in einer Berechnung mit Gleitkommawerten (z.B. Berechnung der Quadratwurzel einer negativen Zahl)	Fehler
Bit 1 = 1	Die Erfassung eines unzulässigen Wertes an einem Gleitkommaeingang kann folgende Ursachen haben: ● der Wert ist kein Gleitkommawert ● der Wert ist unendlich (z.B. Ergebnis einer dem Funktionsbaustein vorgeschalteten Berechnung)	Fehler
Bit 2 = 1	Division durch Null bei Berechnung in Gleitkommawerten	Fehler
Bit 3 = 1	Kapazitätsüberlauf bei Berechnung in Gleitkommawerten	Fehler
Bit 4 = 1	Ein Eingangsparameter ist außerhalb des Bereichs. Der vom Funktionsbaustein intern verwendete Wert wird gekappt.	Warnung oder Information (Hinweis 1)
Bit 5 = 1 (Hinweis 2)	Der Hauptausgang des Funktionsbausteins hat den unteren Grenzwert erreicht	Information
Bit 6 = 1 (Hinweis 2)	Der Hauptausgang des Funktionsbausteins hat den oberen Grenzwert erreicht	Information
Bit 7 = 1	Der untere und obere Grenzwert des Eingabeparameterbereichs sind identisch	Fehler

Hinweis 1 (Eingangs-parameter)

Hinweis: Wenn der Wert aus einem Parameterbereich mit abgeleiteten Datentypen stammt (typischerweise der Parameter PARA), wird aufgrund der Kappung eine Warnung ausgegeben und das Bit 4 auf 1 gesetzt. Wenn der Wert von einem einfachen Eingangstyp stammt, wird keine Warnung ausgegeben, aber Bit 4 des STATUS-Wortes wird auf 1 gesetzt.

Hinweis 2 (Grenzwerte)

Hinweis: Wenn der obere und untere Grenzparameter eines Ausgangs invertiert worden sind (z.B. out_min >= out_max), schaltet der Funktionsbaustein den Ausgang auf den niedrigsten Wert (d.h. auf out_max).

Konvention

Festlegung der Konvention

Wenn ein boolescher Parameter zur Unterscheidung von 2 Betriebsarten oder 2 Zuständen eines Funktionsbausteins verwendet wird, hat sein Name häufig die Form : mode1_mode2 (Beispiel: MANU_AUTO, SP_RSP). Konventionell wird festgelegt, daß der mode1 entsprechende Wert 0 ist und der mode2 entsprechende Wert 1. Wenn z.B. der Parameter MANU_AUTO eines Funktionsbausteins 0 ist, ist der Funktionsbaustein in der Betriebsart Hand. Er ist in der Betriebsart Automatik, wenn MANU_AUTO gleich 1 ist.

EFB-Beschreibungen (A bis PH)

II

Übersicht

Einleitung

Die EFB-Beschreibungen sind in alphabetischer Reihenfolge dokumentiert.

Hinweis: Die Anzahl der Eingänge einiger EFBs kann durch vertikale Größenänderung des FFB-Symbols auf max. 32 erhöht werden. Um welche EFBs es sich dabei handelt entnehmen Sie bitte der Beschreibung der einzelnen EFBs.

Inhalt dieses Teils

Dieser Teil enthält die folgenden Kapitel:

Kapitel	Kapitelname	Seite
3	ALIM: Anstiegsbegrenzer 2. Ordnung	41
4	AUTOTUNE: Automatische Reglereinstellung	47
5	COMP_DB: Vergleich	71
6	COMP_PID: Komplexer PID-Regler	75
7	DEADTIME: Totzeitglied	95
8	DELAY: Totzeitglied	101
9	DERIV: Differenzierer mit Glättung	107
10	DTIME: Verzögerung	113
11	FGEN: Funktionsgenerator	123
12	INTEG: Integrierer mit Begrenzung	139
13	INTEGRATOR: Integrierer mit Begrenzung	145
14	INTEGRATOR1: Integrierer mit Begrenzung	151
15	K_SQRT: Quadratwurzel	157
16	LAG: Verzögerungsglied 1. Ordnung	161
17	LAG1: Verzögerungsglied 1. Ordnung	167
18	LAG2: Verzögerungsglied 2. Ordnung	173
19	LAG_FILTER: Verzögerungsglied 1. Ordnung	179
20	LDLG: PD-Glied mit Glättung	183
21	LEAD: Differenzierer mit Glättung	189
22	LEAD_LAG: PD-Glied mit Glättung	193
23	LEAD_LAG1: PD-Glied mit Glättung	201
24	LIMV: Anstiegsbegrenzer 1. Ordnung	207
25	MFLOW: Baustein für Massendurchfluß	213
26	MS: Handsteuerung eines Ausgangs	219
27	MULDIV_W: Multiplikation/Division	231
28	PCON2: Zweipunkt-Regler	235
29	PCON3: Dreipunkt-Regler	241
30	PD_or_PI: Strukturumschaltung PD- /PI-Regler	249
31	PDM: Pulsdauermodulation	261

ALIM: Anstiegsbegrenzer 2. Ordnung

3

Übersicht

Einleitung

Dieses Kapitel beschreibt den Baustein ALIM.

Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Kurzbeschreibung	42
Darstellung	43
Detailbeschreibung	44
Laufzeitfehler	45

Kurzbeschreibung

Funktionsbeschreibung

Der Funktionsbaustein realisiert einen Anstiegsbegrenzer 2. Ordnung.

Der Funktionsbaustein besitzt im einzelnen folgende Eigenschaften:

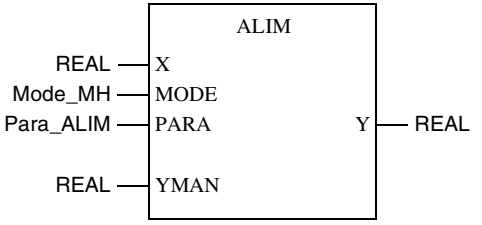
- Betriebsarten Hand, Halt, Automatik
- Ausgangsbegrenzung

Als zusätzliche Parameter können EN und ENO projektiert werden.

Darstellung

Symbol

Darstellung des Bausteins:



Parameterbeschreibung ALIM

Beschreibung der Bausteinparameter:

Parameter	Datentyp	Bedeutung
X	REAL	Eingang
MODE	Mode_MH	Betriebsart
PARA	Para_ALIM	Parameter
YMAN	REAL	Handwert für Ausgang Y
Y	REAL	Ausgang

Parameterbeschreibung Mode_MH

Beschreibung der Datenstruktur:

Element	Datentyp	Bedeutung
man	BOOL	"1" = Betriebsart Hand
halt	BOOL	"1" = Betriebsart Halt

Parameterbeschreibung Para_ALIM

Beschreibung der Datenstruktur:

Element	Datentyp	Bedeutung
max_v	REAL	Maximale Anstiegsgeschwindigkeit (maximales x') Einheit: 1/[s]
max_a	REAL	Maximale Anstiegsbeschleunigung (maximales x'') Einheit: $1/s^2$

Detailbeschreibung

Parametrierung

Die Parametrierung des Funktionsbausteins erfolgt durch Festlegen der maximalen Anstiegsgeschwindigkeit max_v und der maximalen Anstiegsbeschleunigung max_a. Die maximale Anstiegsgeschwindigkeit gibt an, um welchen Wert sich der Ausgang Y innerhalb einer Sekunde ändern darf. Die maximale Anstiegsbeschleunigung gibt an, um welchen Wert sich die Geschwindigkeit des Ausgangs Y maximal ändern darf.

Der Wert von Y folgt dem Wert von X, jedoch begrenzt durch die maximal erlaubte Anstiegsgeschwindigkeit und die maximal erlaubte Anstiegsbeschleunigung.

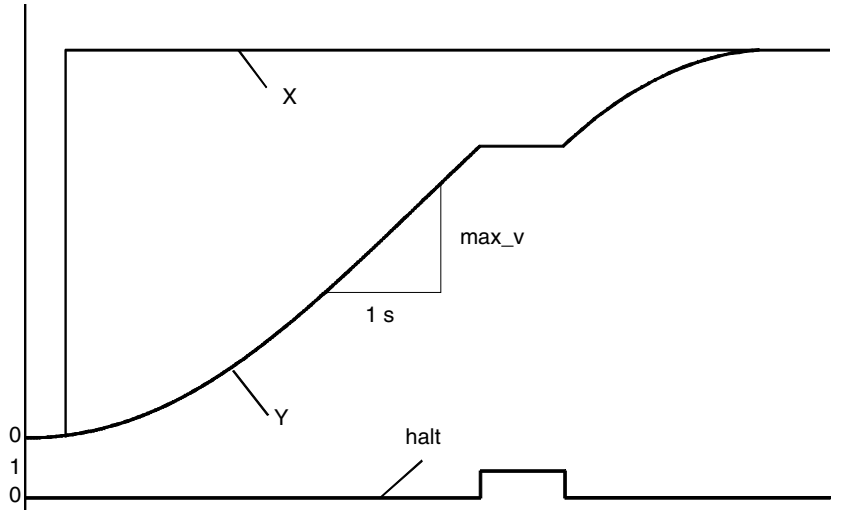
Betriebsarten

Es gibt drei Betriebsarten, die über die Elemente man und halt ausgewählt werden:

Betriebsarten	man	halt	Bedeutung
Automatik	0	0	Es wird ständig ein neuer Wert für Y berechnet und ausgegeben.
Hand	1	0 oder 1	Der Handwert YMAN wird fest auf den Ausgang Y übertragen.
Halt	0	1	Der Ausgang Y wird auf dem zuletzt berechneten Wert gehalten. Der Ausgang wird nicht mehr verändert, kann jedoch vom Anwender überschrieben werden.

Beispiel

Im Bild ist das dynamische Verhalten des Funktionsbausteins sowie die Reaktion bei der Betriebsart HALT dargestellt:



Auf den Sprung am Eingang X reagiert der Funktionsbaustein mit einem beschleunigten Ansteigen des Ausgangs Y. Der Ausgang Y wird mit der im Parameter max_a festgelegten Anstiegsbeschleunigung beschleunigt. Erreicht die Anstiegsgeschwindigkeit den Wert von max_v, so wird nicht weiter beschleunigt, sondern der Ausgang Y folgt dem Eingang X mit der maximalen Anstiegsgeschwindigkeit max_v (siehe Geradenstück in der Mitte des Bildes).

Hat sich der Ausgang Y dem Wert des Eingangssignals genügend genähert, so wird der Ausgang mit der negativen Anstiegsbeschleunigung -max_a gebremst, so daß der Ausgang nicht abrupt stehen bleibt, sondern langsam in den Endwert einläuft.

Laufzeitfehler

Fehlermeldung

Es erfolgt eine Fehlermeldung, wenn

- eine unzulässige Gleitkommazahl am Eingang YMAN oder X anliegt,
- max_a oder max_v ≤ 0 ist.

AUTOTUNE: Automatische Reglereinstellung

4

Übersicht

Einleitung

Dieses Kapitel beschreibt den Baustein AUTOTUNE.

Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Kurzbeschreibung	48
Darstellung	49
Prinzip der automatischen Reglereinstellung	52
Identifikationsprinzip	54
Parametrierung	55
Reglerkopplung	58
Betriebsarten	59
Diagnose	60
Status der automatischen Reglereinstellung	61
Ursachen für einen Fehlstart	61
Ursachen für den Abbruch der automatischen Reglereinstellung	62
Generierung eines Test nach Beenden der automatischen Reglereinstellung	64
Laufzeitfehler	69

Kurzbeschreibung

Funktionsbeschreibung

Dieser Funktionsbaustein ermöglicht die automatische Einstellung der PID-Regler (*PIDFF: Kompletter PID-Regler, S. 351*, *PI_B: Einfacher PI-Regler, S. 291*).

Bei Anlagenstart stabilisiert die automatische Reglereinstellung die Regelung und wirkt somit zeitsparend.

Als zusätzliche Parameter können EN und ENO projektiert werden.

Algorithmus

Der Algorithmus basiert auf heuristischen Regeln wie z.B. die Ziegler-Nichols-Methode. Es wird zunächst eine Analyse durchgeführt, die etwa 2.5 mal der Reaktionszeit der offenen Schleifen entspricht. Hierdurch kann der Prozeß als Prozeß erster Ordnung mit Verzögerung identifiziert werden.

Aufbauend auf diesem Modell wird ein auf heuristischen Regeln und Erfahrungswerten beruhender Regelparametersatz erstellt.

Der Parameterbereich wird durch das Kriterium 'perf' bestimmt. Dieser Faktor erteilt im Einzelfall der Reaktionszeit auf Störungen oder der Stabilität den höchsten Rang zu.

Der Algorithmus wird auf folgende Prozeßtypen angewandt :

- Prozesse mit nur einem Eingang / einem Ausgan
- Prozesse mit natürlicher Stabilität oder mit Integralanteil
- Asymmetrische Prozesse innerhalb der vom Algorithmus des PID-Reglers zulässigen Grenzen
- Prozesse, die über Pulsbreitenmodulationsausgang (PWM) gesteuert werden

Wichtige Merkmale

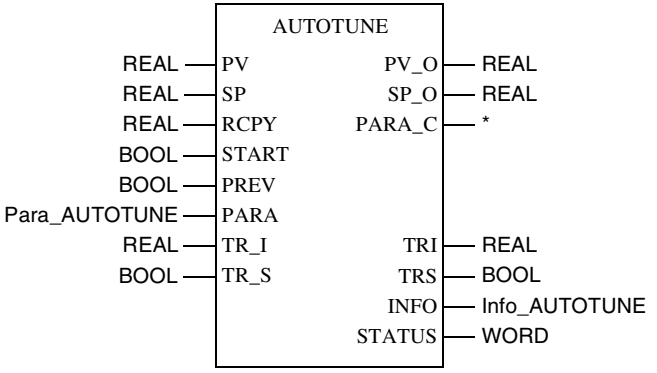
Der Baustein besitzt folgende Merkmale

- Voreinschätzung der Regelung für die Typen PIDFF bzw. PI_B
- Diagnosefunktion
- Parametrierung der Regeldynamik
- Rückgewinnung der vorherigen Regeleinstellungen

Darstellung

Symbol

Darstellung des Bausteins



* Parameter des automatisch eingestellten Reglers (Para_PIDFF, Para_PI_B,... o.a.)

Parameterbeschreibung AUTOTUNE

Beschreibung der Bausteinparameter

Parameter	Datentyp	Bedeutung
PV	REAL	Istwert (Process Value)
SP	REAL	Sollwert (Set Point)
RCPY	REAL	Kopie der tatsächlichen Stellgröße
START	BOOL	"0 → 1" : Starten der automatischen Reglereinstellung
PREV	BOOL	Rückkehr zu den vorherigen Einstellungen des Reglers
PARA	Para_AUTOTUNE	Parameter
TR_I	REAL	Starteingang
TR_S	BOOL	Startbefehl
PV_O	REAL	Kopie des Istwertes PV
SP_O	REAL	Kopie des SP-Eingangs
PARA_C	Parameter des automatisch einstellbaren Reglers (Para_PIDFF bzw. Para_PI_B)	Regelparameter
TRI	REAL	Kopie des Eingangs TR_I
TRS	BOOL	Kopie des Eingangs TR_S
INFO	Info_AUTOTUNE	Information
STATUS	WORD	Statuswort

Parameterbeschreibung Para_ AUTOTUNE

Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
step_ampl	REAL	Wert des Ausgangsstellimpulses (ausgedrückt in Ausgangsskalierwerten out_inf, out_sup)
tmax	TIME	Dauer des Stellimpulses bei autom. Reglereinstellung
perf	REAL	Leistungsindex zwischen 0 und 1
plant_type	WORD	Reserviertes Wort

Parameterbeschreibung Info_AUTOTUNE

Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
diag	UDINT	Für Diagnose verwendetes Doppelwort
p1_prev	REAL	Vorheriger Wert von Parameter 1
p2_prev	REAL	Vorheriger Wert von Parameter 2
p3_prev	REAL	Vorheriger Wert von Parameter 3
p4_prev	REAL	Vorheriger Wert von Parameter 4
p5_prev	REAL	Vorheriger Wert von Parameter 5
p6_prev	REAL	Vorheriger Wert von Parameter 6

Prinzip der automatischen Reglereinstellung

Zwei Arten der automatischen Regler-einstellung

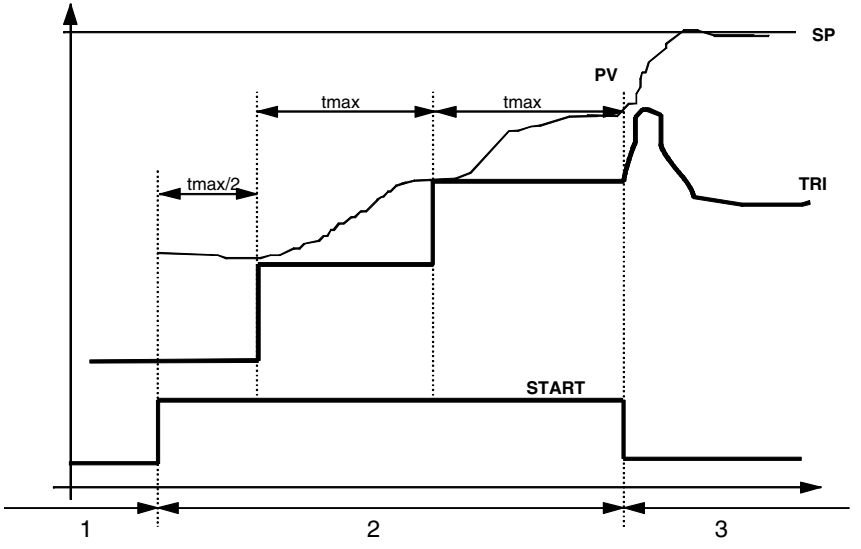
Zwei Arten der automatischen Reglereinstellung sind möglich: automatische Reglereinstellung bei Warmstart und bei Kaltstart des Systems.

Die erste Phase der automatischen Reglereinstellung gilt für beide Einstellungsarten : hierbei handelt es sich um einen Rausch- und Stabilitätstest des Regelprozesses mit einer Dauer von 0.5 * tmax bei konstanten Ausgängen. Die nachfolgenden Phasen sind abhängig von der Einstellart.

Automatische Regler-einstellung bei Kaltstart

Man spricht von der automatischen Reglereinstellung bei Kaltstart, wenn die Abweichung zwischen dem Ist- und dem Sollwert 40% überschreitet und der Istwert 30 % unterschreitet. In diesem Falle wird der Ausgang TRI des Funktionsbausteins mit zwei gleichartigen Stellimpulsen beaufschlagt. Jeder Stellimpuls hat eine Dauer von tmax. Bei Beendigung der Reglereinstellung wird zur dem Regelkreis vorausgehenden Betriebsart zurückverschleift:

Automatische Reglereinstellung bei Kaltstart

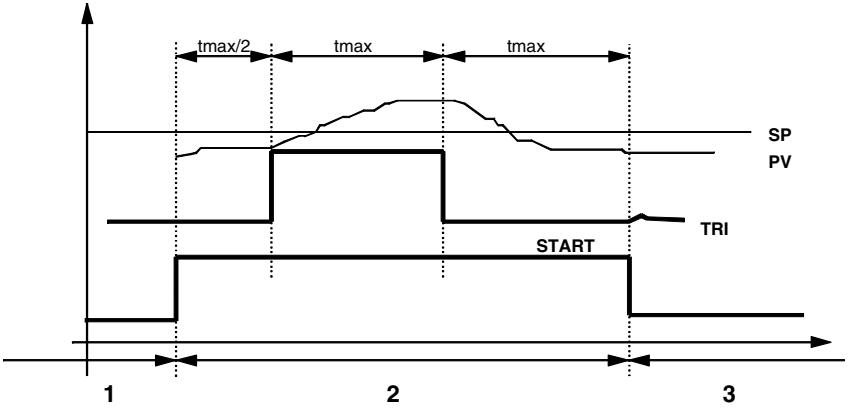


1 Betriebsart Automatik oder Hand
2 Betriebsart automatische Reglereinstellung
3 Betriebsart Automatik oder Hand

Automatische Regler-einstellung bei Warmstart

Wenn die Bedingungen für eine automatische Reglereinstellung bei Kaltstart nicht erfüllt sind, erfolgt eine Einstellung bei Warmstart: der Ausgang wird mit einem Stellimpuls beaufschlagt, gefolgt von einem Stellimpuls in entgegengesetzter Richtung. Jede Etappe hat eine Dauer von tmax. Wenn die automatische Reglereinstellung beendet ist, wird zur dem Regelkreis vorausgegangenen Betriebsart zurückverschleift:

Automatische Reglereinstellung bei Warmstart



1 Betriebsart Automatik oder Hand
2 Betriebsart automatische Reglereinstellung
3 Betriebsart Automatik oder Hand

Identifikationsprinzip

Identifikations-prozeß

Der Identifikationsprozeß umfaßt 3 Etappen :

- eine Rausch- und Stabilitätsanalyse des Regelprozesses
- eine erste Analyse der Reaktion auf einen Stellimpuls, die das erste Identifikationsmodell darstellt: aufgrund dieser ersten Einschätzung wird ein Filter erstellt, das bei der letzten Phase verwendet wird
- eine zweite Analyse der Reaktion auf einen zweiten Stellimpuls liefert aufgrund des Datenfilters präzisere Angaben

Schließlich wird ein vollständiges Prozeßmodell erstellt. Wenn die Ergebnisse der beiden vorherigen Phasen zu weit auseinander liegen, wird die Schätzung verworfen und die automatische Reglereinstellung scheitert.

Regelprinzip

Nach jeder der beiden Phasen wird ein Parametersatz für den einzustellenden Regler erstellt. Die die Regelparameter ergebenden Gleichungen basieren auf dem Verstärkungsfaktor und dem Verhältnis zwischen Reaktionszeit und Prozeß-Verzögerung.

Der Algorithmus muß der Änderung des Verstärkungsfaktors und der Zeitkonstante im Verhältnis 2 ohne Stabilitätsverlust standhalten können. Die asymmetrischen Prozesse werden unterstützt, wenn sie diese Bedingung erfüllen. Wenn dies nicht der Fall ist, wird bei Diagnose diag. ein Fehler angezeigt.

Parametrierung

Parametrierung des Stellimpulses

Bei der automatischen Reglereinstellung werden dem Ausgang TRI zwei Stellimpulse aufgeschaltet. Ein Stellimpuls ist durch zwei Parameter gekennzeichnet: seine Dauer (tmax) und seine Amplitude (step_ampl).

Für diese Parameter gelten folgende Wertebereiche: tmax größer als 4 Sekunden und step_ampl größer als 1 % der Ausgangsskala (out_inf, out_sup). Die Funktion überwacht ebenfalls, ob der TRI-Ausgang die Grenzwerte der Ausgangsskala nicht überschreitet

Die Prüfung erfolgt bei Start der automatischen Reglereinstellung.

Folgende Tabelle enthält Parameterwerte für einige typische Regelarten:

Schema	**tmax (s)**	**step_ampl (%)**
Durchflußvol. oder Druck von Flüssigkeiten	5-30	10-20
Gasdruck	60-300	10-20
Pegel	120-600	20
Dampftemperatur oder -druck	600-3600	30-50
Zusammensetzung	600-3600	30-50

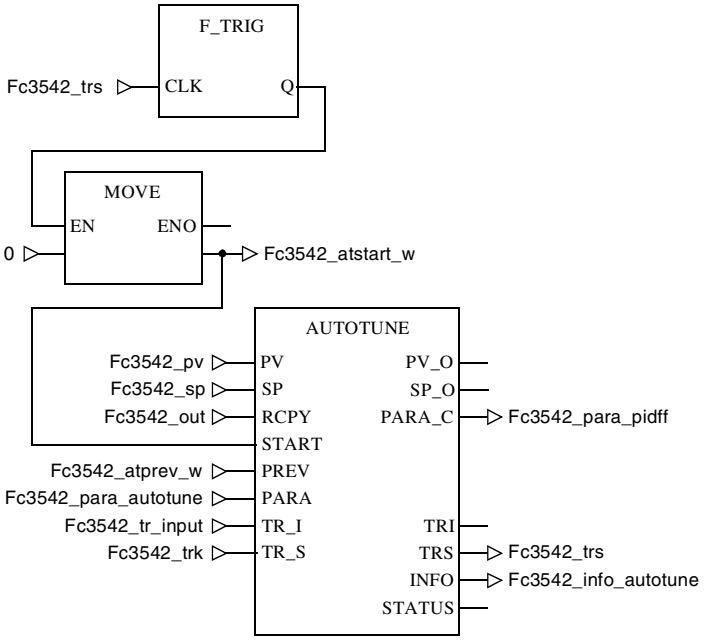
Leistungsindex: perf

Der Regler kann je nach Wert des Leistungsindexes moduliert werden. Der Leistungsindex perf variiert zwischen 0 und 1, was es ermöglicht, den Parameter perf nahe 0 zu stabilisieren oder eine dynamischere Regelung zu erzielen (und somit die Reaktionszeit auf Störgrößen zu optimieren), wenn perf nahe 1 eingestellt ist.

Starten der automatischen Reglerein-stellung: START

Wenn dieses Bit auf 1 gesetzt wird, wird die Funktion aktiviert. Am Ende des Einstellvorgangs muß dieses Bit durch den Anwender auf Null gesetzt werden. Wenn gerade automatisch eingestellt wird, ermöglicht die Nullsetzung des Bits, die Funktion zu stoppen. Die PARA_C Parameter behalten dann den letzten noch aktiven Wert bei. Im unten gezeigten Beispiel wird das Bit START durch das Programm automatisch am Ende des Einstellvorgangs zurückgesetzt.

Beispiel zum Starten der automatischen Reglereinstellung



Rückkehr zur vorherigen Einstellung: PREV

Eine Änderung dieses Bitwertes ermöglicht die Vertauschung der aktuellen und der vorherigen Parameter unter der Voraussetzung, daß zum gegebenen Zeitpunkt keine Regelung stattfindet (zwei aufeinanderfolgende Änderungen dieses Bits ergeben folglich die anfängliche Konfiguration).

Beim Regler des Typs PIDFF gelten folgende Info_AUTOTUNE Strukturparameter:

Element der Datenstruktur	Bedeutung
p1_prev	KP
p2_prev	TI
p3_prev	TD

Beim Regler des Typs PI_B gelten folgende Info_AUTOTUNE Strukturparameter:

Element der Datenstruktur	Bedeutung
p1_prev	KP
p2_prev	TI

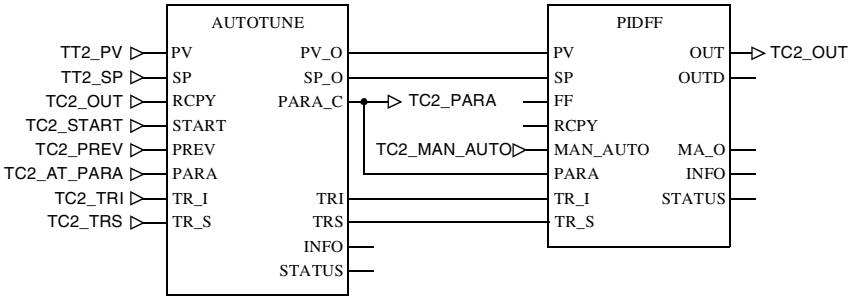
Diagnose bei automatischer Reglerein-stellung: diag

Die Diagnosedaten der automatischen Reglereinstellung werden in einem Doppelwort gespeichert. Der Wert dieses Wortes bleibt bis zu einem erneuten Start der automatischen Reglereinstellung erhalten. Weitere Angaben zu diesem Doppelwort finden Sie im Abschnitt Diagnose.

Reglerkopplung

Anwendungsbeispiel mit einem EFB des Reglertyps PIDFF

Folgendes Diagramm ist ein Anwendungsbeispiel eines AUTOTUNE EFBs mit einem EFB des Reglertyps PIDFF :



Der AUTOTUNE EFB tauscht mit dem Regler Parameter aus: Der Zugang zu den Reglerparametern erfolgt über die Verbindung zwischen dem Ausgang PARA_C des AUTOTUNE-Funktionsbausteins und dem Eingang PARA des Reglers. Der PARA_C-Ausgang ist vom Typ ANY und ermöglicht den Anschluß des AUTOTUNE-EFBs an verschiedene Reglertypen (PIDFF oder PI_B).

Darüber hinaus teilen sich der AUTOTUNE EFB und der Regler folgende verkettbare Variablen: PV, SP, TR_I und TR_S. Diese Variablen stellen AUTOTUNE-Eingänge dar, die auf die entsprechenden Ausgänge geführt werden, um die Aufschaltung auf die Reglereingänge zu ermöglichen

Wenn die automatische Reglereinstellung aktiv ist, geht der TRS-Ausgang auf 1 über und die Stellgröße wird an den TRI-Ausgang angelegt. Sinn dieser Ausgänge ist der Anschluß an die Eingänge TR_I und TR_S der auf AUTOTUNE folgenden Funktionsbausteine. Diese können so auf die Betriebsart Tracking eingestellt werden (PIDFF, PI_B, MS,...).

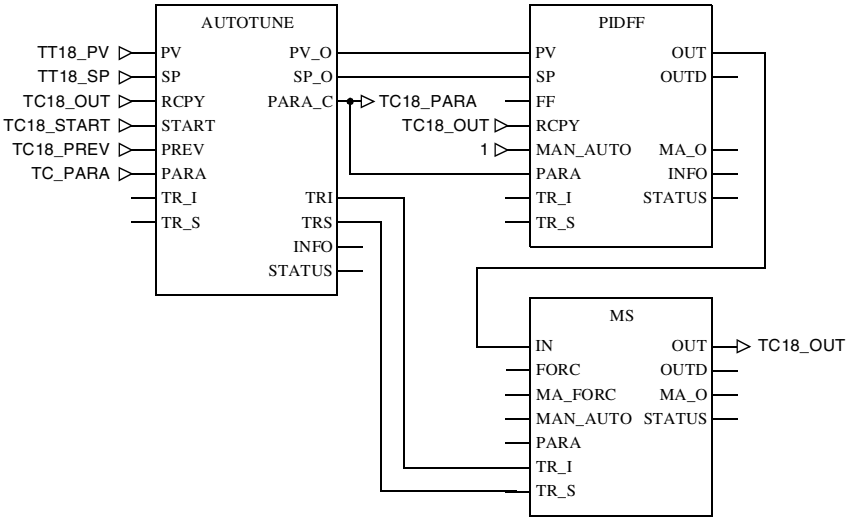
Anschlußbeispiel: Regelkreis mit einem einfachen PID-Regler

In diesem Abschnitt beschäftigen wir uns mit der automatischen Einstellung eines einzelnen Reglers (häufigster Fall). Der Regler kann vom Typ PI_B oder PIDFF sein.

Der AUTOTUNE EBF benötigt die Skalierparameter des Reglers (PARA_C Strukturparameter) pv_inf, pv_sup, out_inf, out_sup sowie den Strukturtyp des Reglers, der durch das Bit mix_par angegeben wird. Der EFB erstellt daraus die Parameter des PID-Reglers (KP, TI, TD). Die Wirkrichtung des Reglers (rev_dir) wird beim Test der automatischen Reglereinstellung geprüft und mit dem Vorzeichen der Verstärkungsfaktoren des Modells verglichen. Bei Inkompatibilität wird für den "diag."-Parameter ein Fehler angezeigt.

Anschluß-beispiel: Regelkreis mit einfachem PID-Regler und MS-Funktions-baustein

Wenn der Regelkreis ein MS-EFB enthält, kann das Strukturbild wie folgt aussehen:



Beim Start der automatischen Reglereinstellung stellt der AUTOTUNE EFB den Funktionsbaustein MS auf die Betriebsart Tracking ein und steuert so direkt den Ausgang des Regelkreises. Die Verwendung der RCPY-Eingänge der Bausteine AUTOTUNE und PIDFF ermöglicht einen stoßfreien Weiterstart des Regelkreises.

Betriebsarten

Betriebsarten

Die verschiedenen Betriebsarten der automatischen Reglereinstellung und deren untereinander geltende Prioritäten sind in folgender Tabelle definiert:

Betriebsart	TR_S	START
Tracking	1	0 oder 1
Automatische Reglereinstellung	0	1

Bei Beendigung der automatischen Reglereinstellung wird der TRS Ausgang auf Null gesetzt und generiert somit eine Rückstellung des Regelkreises auf die vorherige Betriebsart (Hand oder Automatik). Wenn die automatische Reglereinstellung scheitert, wird die Variable TRI auf den vor Start der automatischen Reglereinstellung geltenden Wert zurückgesetzt und der Regelkreis kehrt zur vorherigen Betriebsart zurück.

Diagnose

Übersicht zur Diagnose

Verschiedene Gründe können dazu führen, daß die automatische Reglereinstellung nicht gestartet werden kann, abgebrochen wird oder scheitert. Dabei kann eventuell, je nach Grund des Scheiterns, ein Parametersatz angeboten werden. Jedes Bit des Diagnosewortes diag ermöglicht die Erstellung eines Fehlertyps.

Dieses Wort beinhaltet die aktuelle Betriebsart der automatischen Reglereinstellung.

Die folgenden Fälle werden erläutert:

- *Status der automatischen Reglereinstellung, S. 61*
- *Ursachen für einen Fehlstart, S. 61*
- *Ursachen für den Abbruch der automatischen Reglereinstellung, S. 62*
- *Generierung eines Test nach Beenden der automatischen Reglereinstellung, S. 64*

Diagnosewort

IIn dieser Tabelle finden Sie die Bedeutung des Elements diag der Datenstruktur Info_AUTOTUNE

Bit	Bedeutung
Bit 0 = 1	Automatische Reglereinstellung läuft
Bit 1 = 1	Reglereinstellung abgebrochen
Bit 2 = 1	Parameterfehler
Bit 3 = 1	Änderung von Parametern, die gerade automatisch eingestellt werden
Bit 4 = 1	Stop infolge Systemfehler
Bit 5 = 1	Sättigung des Istwertes
Bit 6 = 1	Änderung zu gering
Bit 7 = 1	Abtastintervall ungültig
Bit 8 = 1	Reaktion unverständlich
Bit 9 = 1	Anfänglich nicht stabilisierte Messung
Bit 10 = 1	Dauer des Stellimpulses (tmax) zu kurz
Bit 1 1= 1	Zu starkes Meßrauschen
Bit 12 = 1	Stellimpulsdauer (tmax) zu groß
Bit 13 = 1	Prozeß mit starker Überschreitung der Grenzwerte
Bit 14 = 1	Prozeß ohne Phasenminimum
Bit 15 = 1	Asymmetrischer Prozeß
Bit 16 = 1	Prozeß mit Integralanteil

Status der automatischen Reglereinstellung

Übersicht

Folgende Bits des Diagnosewortes (Element diag) zeigen den Status der automatischen Reglereinstellung

Bit	Bedeutung
0	1 = Automatische Reglereinstellung läuft
1	1 = Automatische Reglereinstellung gestoppt

Bit 0 des Elementes diag

Dieses Bit zeigt an, daß die Reglereinstellung läuft. Bei Beendigung der Reglereinstellung oder Abbruch durch das START-Bit wird dieses auf Null gesetzt.

Bit 1 des Elementes diag

Dieses Bit gibt an, daß der Anwender die letzte Regelung über das START-Bit oder durch Einstellung auf die Betriebsart Tracking gestoppt hat.

Ursachen für einen Fehlstart

Übersicht

Folgende Bits des Diagnosewortes (Element diag) zeigen einen Fehlstart an:

Bit	Bedeutung
2	1 = Parameterfehler
7	1 = Abtastintervall nicht korrekt

Bit 2 des Elementes diag

Folgende Gründe können zu einem Fehlstart führen :

- Dauer des Stellimpulses zu schwach (tmax < 4 s),
- Amplitude zu schwach (step_ampl < 1% des Ausgangsbereichs),
- Protokoll nicht durchführbar: wenn der aktuelle Ausgang + n x Amplitude des Stellimpulses (n = 1 bei Einstellung bei Warmstart und n = 2 bei Einstellung bei Kaltstart) sich außerhalb des Ausgangsbereichs befindet (out_inf, out_sup), kann das Testprotokoll nicht verwendet werden. Der Einstellwert von step_ampl muß kompatibel sein zum aktuellen Arbeitspunkt.

Bit 7 des Elementes diag

Wenn das Abtastintervall im Verhältnis zur Stellimpulsdauer zu groß ist (> tmax / 25), ist die Reaktionsprüfung zu unpräzise und die automatische Reglereinstellung wird blockiert. Dieser Fall ist typisch für sehr schnelle Regelprozesse (tmax größer als Einschwingzeit des Prozesses, einer Größenordnung von wenigen Sekunden). In diesem Fall kann tmax erhöht werden, da der Algorithmus kaum auf diesen Parameter reagiert (im Verhältnis 1 zu 3) oder das Abtastintervall entsprechend eingestellt werden.

Ursachen für den Abbruch der automatischen Reglereinstellung

Übersicht

Folgende Bits des Diagnosewortes (Element diag) zeigen die Ursachen für den Abbruch der automatischen Reglereinstellung:

Bit	Bedeutung
3	1 = Änderung von Parametern während der Reglereinstellung
4	1 = Abbruch wegen Systemfehler
5	1 = Sättigung des Istwertes
6	1 = Anstieg zu gering
8	1 = Unlogische Reaktion

Bit 3 des Elementes diag

Wenn der Parameter tmax oder step_ampl während der Reglereinstellung geändert wird, wird diese abgebrochen.

Bit 4 des Elementes diag

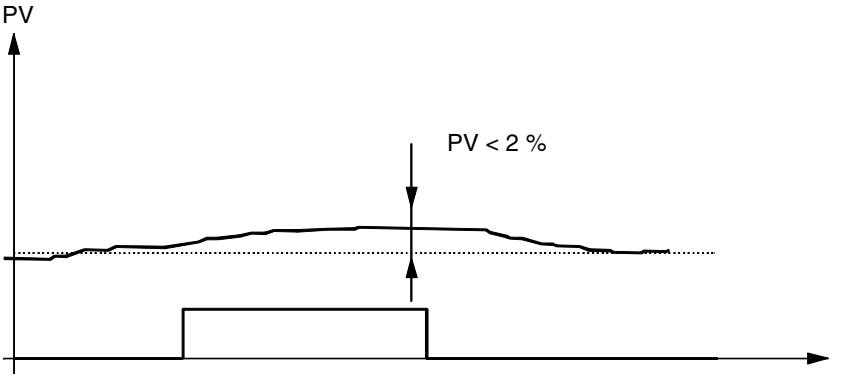
Die automatische Reglereinstellung wird abgebrochen, wenn an der SPS ein Systemfehler vorliegt, der den vollständigen Ablauf der Ablaufkette behindert. Ein Ausschalten stoppt beispielsweise automatisch die Funktion bei Spannungsrückkehr.

Bit 5 des Elementes diag

Wenn der Meßwert den Meßbereich überschreitet (pv_inf, pv_sup), wird die automatische Reglereinstellung abgebrochen und der Regler auf die vorherige Betriebsart eingestellt. Eine Schätzung des künftigen Meßwertes ermöglicht, die automatische Reglereinstellung zu stoppen, bevor der Meßbereich überschritten wird (wenn ein erstes Modell identifiziert worden ist).

Bit 6 des Elementes diag

Dieses Bild zeigt das Verhalten bei zu geringem Anstieg:



Die Amplitude des Stellimpulses ist zu gering, um den Prozeß zu beeinflussen. In diesem Fall kann der Wert von step_ampl erhöht werden.

Bit 8 des Elementes diag

Dieses Bild zeigt das Verhalten bei unlogischer Reaktion:

Die Reaktion des Regelprozesses ist unverständlich (Verstärkungsfaktoren mit verschiedenen Vorzeichen). Dies kann auf eine größere Störung, auf eine Kopplung mit anderen Regelkreisen o.a. zurückzuführen sein.

Generierung eines Test nach Beenden der automatischen Reglereinstellung

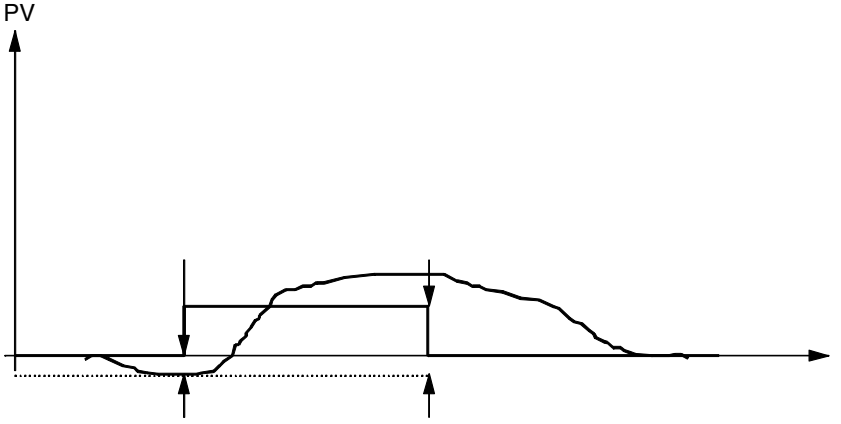
Übersicht

Folgende Bits des Diagnosewortes (Element diag) zeigen den Status der automatischen Reglereinstellung:

Bit	Bedeutung
9	1 = Anfänglich nicht stabilisierter Meßwert
10	1 = Dauer des Stellimpulses (tmax) zu kurz
11	1 = Zu hohes Meßrauschen
12	1 = Dauer des Stellimpulses (tmax) zu lang
13	1 = Starke Überschreitung des Meßwertes
14	1 = Prozeß ohne Phasenminimum
15	1 = Asymmetrischer Prozeß
16	1 = Integrierender Prozeß

Bit 9 des Elementes diag

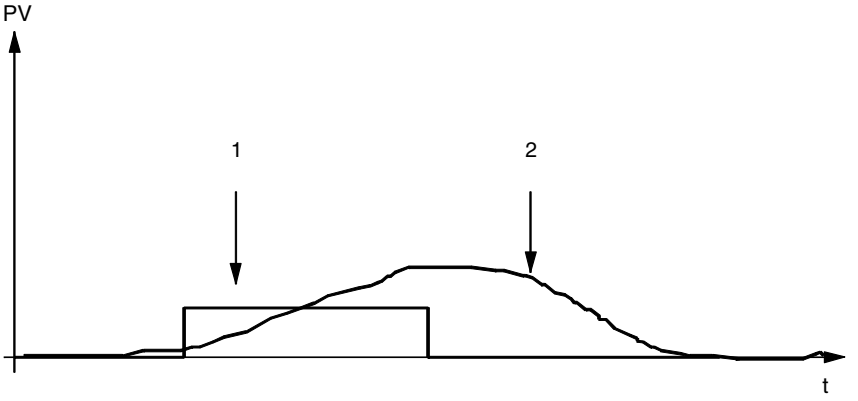
Dieses Bild zeigt das Verhalten bei anfänglich nicht stabilisiertem Meßwert:



Die automatische Reglereinstellung wurde durchgeführt, obwohl der Meßwert nicht stabil war. Wenn die Meßwertänderung im Verhältnis zur Reaktion des Stellimpulses groß ist, werden die Testergebnisse verfälscht.

Bit 10 des Elementes diag

Dieses Bild zeigt das Verhalten bei zu geringer Dauer des Stellimpulses:

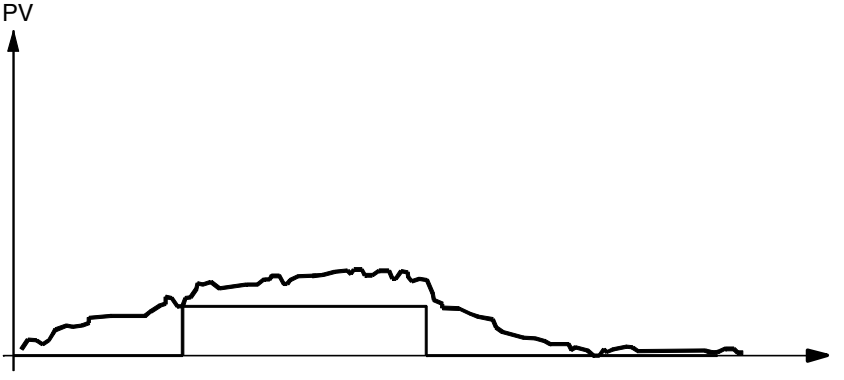


1 Test-Stellimpuls
2 Reaktion des Prozesses

Die Reaktion wird nicht vor Rückkehr zur anfänglichen Stellgröße stabilisiert. Die berechneten Parameter sind daher falsch.

Bit 11 des Elementes diag

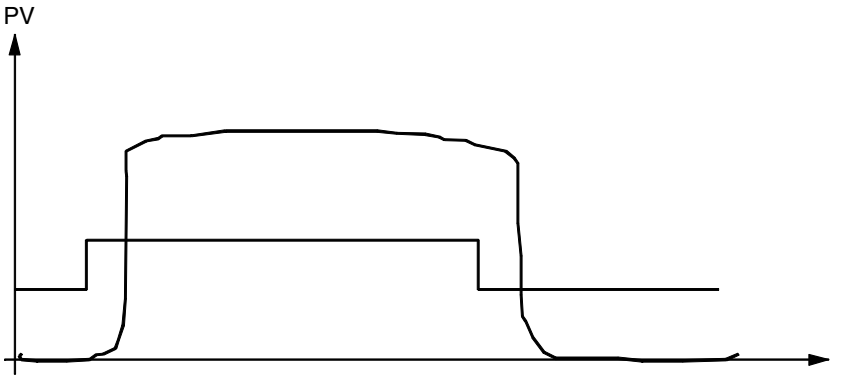
Dieses Bild zeigt das Verhalten bei zu hohem Meßrauschen:



Die Reaktion des Prozesses auf den Stellimpuls ist im Verhältnis zum Meßrauschpegel nicht ausreichend. Der Meßwert muß gefiltert oder step_ampl erhöht werden.

Bit 12 des Elementes diag

Dieses Bild zeigt das Verhalten bei zu langer Dauer des Stellimpulses:



tmax bestimmt die Häufigkeit der Meßwerterfassung, d.h. der Werte, die zur Berechnung der Koeffizienten dienen. tmax muß zwischen 1 und 5 mal der Anstiegszeit des Regelprozesses entsprechen.

Bit 13 des Elementes diag

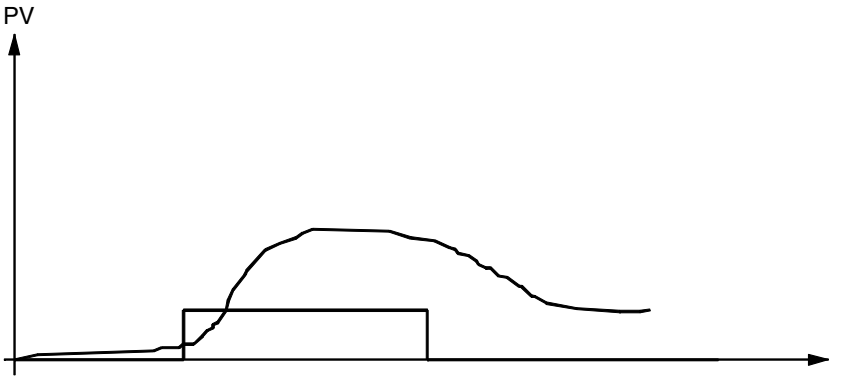
Dieses Bit wird gesetzt, wenn die Reaktion auf einen Stellimpuls zu einer zu großen Überschreitung des Meßwertes (overshoot) führt (mehr als 10%). Der Prozeß entspricht nicht den vom Algorithmus verarbeiteten Modellen.

Bit 14 des Elementes diag

Dieses Bit wird gesetzt, wenn die Reaktion auf einen Stellimpuls zu einer Reaktionsumkehr in der Anfangsphase führt (undershoot größer als 10%). Der Prozeß entspricht nicht den vom Algorithmus verarbeiteten Modellen.

Bit 15 des Elementes diag

Dieses Bild zeigt das Verhalten bei assymetrischem Prozess:



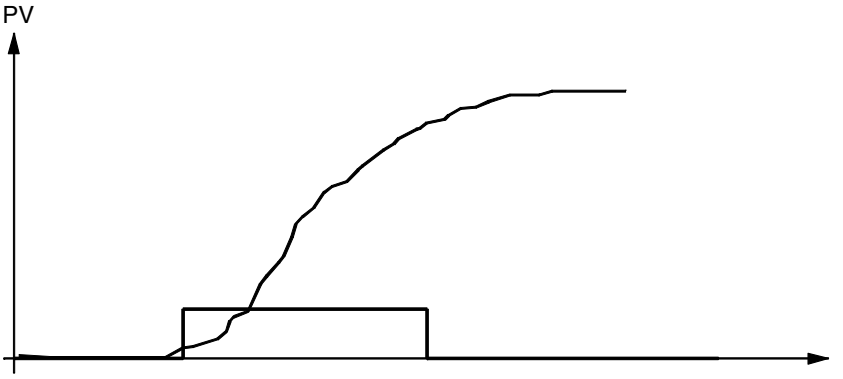
Die Reaktion des Prozesses ist asymmetrisch.

Der Endparametersatz entspricht einem Kompromiß zwischen den Reaktionen bei An- und Abstieg. In beiden Fällen handelt es sich um mittlere Leistungen.

Wenn das gewünschte Kriterium die Reaktionsdauer auf den Anstieg ist, ist der erste Parametersatz zu berücksichtigen. Dabei wird in der Rückkehrphase zur anfänglichen Stellgröße die automatische Reglereinstellung abgeschaltet. Wenn das gewünschte Kriterium die Abstiegsdauer ist, ist eine negative Amplitude zu verwenden.

Bit 16 des Elementes diag

Dieses Bild zeigt das Verhalten bei integrierendem Prozess:



Der Prozeß enthält einen Integralanteil oder tmax ist zu klein und der Prozeß asymmetrisch. Die berechneten Koeffizienten entsprechen dem Prozeß mit Integralbeiwert. Wenn dies nicht der Fall ist, ist die automatische Reglereinstellung neu zu starten, nachdem tmax erhöht wurde.

Laufzeitfehler

Statuswort

Die Bits des Statuswortes haben folgende Bedeutung:

Bit	Bedeutung
Bit 0 = 1	Fehler in einer Berechnung mit Gleitkommawerten
Bit 1 = 1	Erfassung eines unzulässigen Wertes an einem der Gleitkommaeingänge
Bit 2 = 1	Division durch Null bei Berechnung in Gleitkommawerten
Bit 3 = 1	Kapazitätsüberlauf bei Berechnung in Gleitkommawerten
Bit 4 = 1	Der Parameter perf liegt außerhalb des Bereichs [0,1]: Der Funktionsbaustein verwendet zur Berechnung den Wert 0 oder 1
Bit 7 = 1	Die Grenzwerte pv_inf und pv_sup des einzustellenden Reglers sind identisch
Bit 8 = 1	Der Ausgang PARA_C ist nicht an die Parameter eines automatisch einstellbaren Reglers angeschlossen
Bit 9 = 1	Automatische Reglereinstellung gescheitert
Bit 10 = 1	Die letzte automatische Reglereinstellung war erfolgreich

Fehlermeldung

Dieser Fehler wird angezeigt, wenn an einem Eingang ein nicht gleitender Wert festgestellt wird, bei einer Berechnung in Gleitkommawerten ein Problem auftritt oder die Grenzwerte pv_inf und pv_sup des Reglers identisch sind. In diesem Fall bleiben alle Ausgänge des Funktionsbausteins unverändert.

Warnung

Es wird eine Warnung ausgegeben, wenn der Parameter perf außerhalb des Bereichs [0,1] liegt. Der Baustein kann in diesem Fall zur Berechnung sowohl den Wert 0, wie auch den Wert 1 verwenden.

COMP_DB: Vergleich

5

Übersicht

Einleitung

Diese Kapitel beschreibt den Baustein COMP_DB

Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Kurzbeschreibung	72
Darstellung	72
Detailbeschreibung	73
Laufzeitfehler	74

Kurzbeschreibung

Funktionsbeschreibung

Der Funktionsbaustein COMP_DB ermöglicht den Vergleich zweier numerischen Werte IN1 und IN2.

Je nachdem, ob IN1 größer, gleich oder kleiner ist als IN2, wird durch den Funktionsbaustein einer der Ausgänge GREATER, EQUAL oder LESS auf 1 gesetzt.

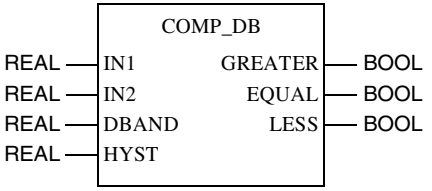
Der Funktionsbaustein berücksichtigt ebenfalls eine Unempfindlichkeitszone und eine Hysterese.

Als zusätzliche Parameter können EN und ENO projektiert werden.

Darstellung

Symbol

Darstellung des Bausteins



Parameterbeschreibung

Beschreibung der Bausteinparameter

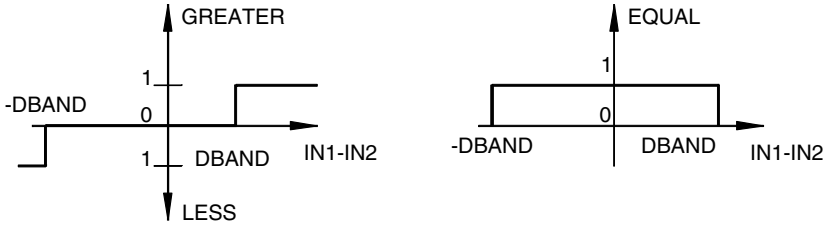
Parameter	Datentyp	Bedeutung
IN1	REAL	Eingang Nr. 1
IN2	REAL	Eingang Nr. 2
DBAND	REAL	Unempfindlichkeitszone
HYST	REAL	Hysterese
GREATER	BOOL	Größer-Merker
EQUAL	BOOL	Gleich-Merker
LESS	BOOL	Kleiner-Merker

Detailbeschreibung

Unempfindlichkeitszone

Der Parameter DBAND ermöglicht die Festlegung einer Unempfindlichkeitszone, in der die Abweichung zwischen IN1 und IN2 als Null angesehen wird. Wenn der Betrag der Abweichung IN1 - IN2 innerhalb dieser Zone bleibt, wird der Ausgang EQUAL auf 1 gesetzt.

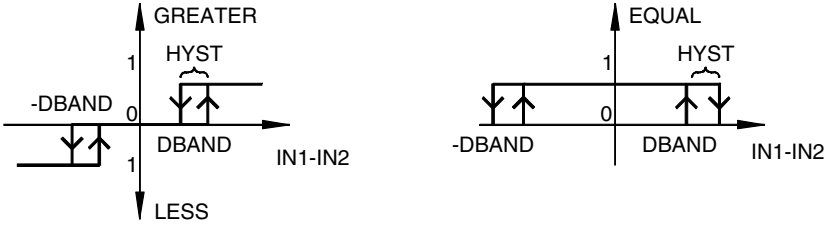
Festlegung der Unempfindlichkeitszone



Hysterese

Der Parameter HYST ermöglicht die Generierung eines Hystereseeffektes, wenn die Abweichung zwischen IN1 und IN2 abnimmt: ausgehend von einem Zustand, wo der Ausgang GREATER oder LESS den Wert 1 annimmt, nimmt der Ausgang EQUAL nur dann den Wert 1 an, wenn die Abweichung IN1 - IN2 betragsmäßig kleiner ist als DBAND - HYST

Generierung eines Hystereseeffektes

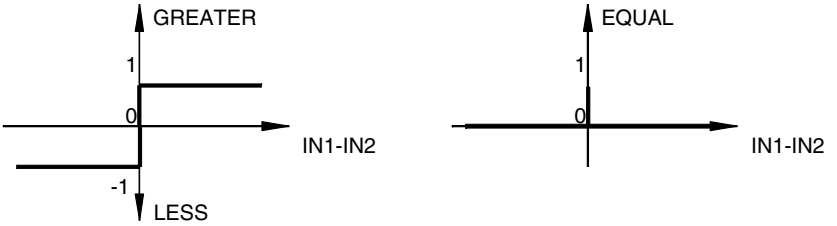


DBAND = 0 und HYST = 0

In diesem Fall verhält sich der Baustein wie eine klassische Vergleichsfunktion:

- Wenn IN1 durchgehend größer ist als IN2, gilt GREATER = 1
- Wenn IN1 gleich IN2 ist, gilt EQUAL = 1
- Wenn IN1 kleiner ist als IN2, gilt LESS = 1

Klassische Vergleichsfunktion (DBAND = 0 und HYST = 0



Laufzeitfehler

Fehlermeldung

Dieser Fehler erscheint, wenn an einem Eingang ein nicht gleitender Wert erfaßt wird oder wenn bei einer Berechnung in Gleitkommawerten ein Problem auftritt. In diesem Fall bleiben die Ausgänge GREATER, EQUAL und LESS unverändert.

Warnung

Eine Warnung wird ausgegeben, wenn:

- Der Parameter DBAND negativ ist: der Funktionsbaustein verwendet dann zur Berechnung DBAND = 0.
- Der Parameter HYST außerhalb des Bereichs [0, DBAND] liegt: zur Berechnung verwendet der Funktionsbaustein den nächstliegenden korrekten Wert, d.h. 0, wenn HYST kleiner ist als 0 und DBAND, wenn HYST größer ist als DBAND.

COMP_PID: Komplexer PID-Regler

6

Übersicht

Einleitung

Diese Kapitel beschreibt den Baustein COMP_PID.

Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Kurzbeschreibung	76
Darstellung	77
Strukturbild des komplexen PID-Reglers	80
Parametrierung des COMP_PID-Reglers	81
Antiwindup für COMP_PID	84
Auswahl des Reglertyps für COMP_PID	85
Stoßfreie Betriebsartenumschaltung	86
Auswahl der Betriebsart des COMP_PID	89
Detaillierte Formeln	92
Laufzeitfehler	94

Kurzbeschreibung

Funktionsbeschreibung

Der Funktionsbaustein stellt einen komplexen PID-Regler dar, der speziell auch für Kaskadenbehandlung konzipiert wurde. Die Struktur des Reglers ist im *Strukturbild, S. 80* dargestellt.

Als zusätzliche Parameter können EN und ENO projektiert werden.

Eigenschaften

Der Funktionsbaustein besitzt folgenden Eigenschaften:

- realer PID-Regler mit unabhängiger gain, ti, td-Einstellung
- Betriebsarten Hand, Halt, Automtik, Kaskade, Rücksetzten, Handwert nachführen
- Anstiegsbegrenzung für Handbetrieb
- Nachführung des Handstellwertes einstellbar
- Anstiegsbegrenzung für Führungsgröße
- stoßfreie Umschaltung Hand/Automatik
- Stellgrößen-Begrenzung
- stoßfrei, einzeln zuschaltbarer P-, I- und D-Anteil
- stoßfreie Änderung der Verstärkung
- Antiwindup-Reset oder Antiwindup-Halt wählbar
- Verschiebung der Antiwindup-Grenzen gegenüber den Stellgrenzen
- Antiwindup-Maßnahme nur bei aktivem I-Anteil
- definierbare Verzögerung des D-Anteils
- D-Anteil umschaltbar auf Regelgröße PV oder Regeldifferenz ERR
- Unempfindlichkeitszone mit Verstärkungsreduktion
- externer Arbeitspunkt (im P-, PD- und D-Betrieb)
- Umschaltung Hand/Automatik wählbar mit Stoß oder stoßfrei

Übertragungsfunktion

Die Übertragungsfunktion lautet:

$$G(s) = \text{gain} \times \left(1 + \frac{1}{\text{ti} \times s} + \frac{\text{td} \times s}{1 + \text{td_lag} \times s}\right)$$

YD
YI
YP

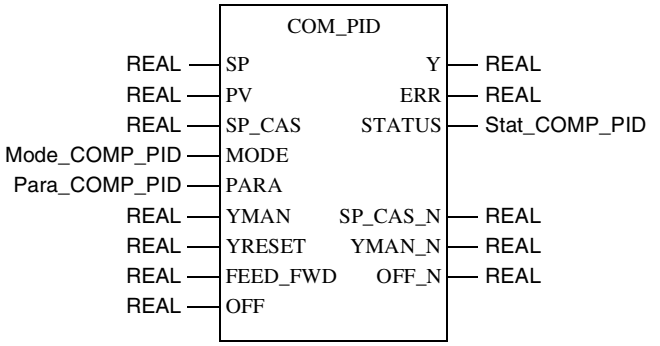
Erläuterung der Größen

Größe	Bedeutung
YD	D-Anteil (nur bei en_d=1)
YI	I-Anteil (nur bei en_i=1)
YP	P-Anteil (nur bei en_p=1)

Darstellung

Symbol

Darstellung des Bausteins:



Parameterbeschreibung COMP_PID

Beschreibung der Bausteinparameter

Parameter	Datentyp	Bedeutung
SP	REAL	Führungsgröße
PV	REAL	Regelgröße
SP_CAS	REAL	Führungsgröße für Kaskade
MODE	Mode_COMP_PID	Betriebsart
PARA	Para_COMP_PID	Parameter
YMAN	REAL	Hand-Stellwert
YRESET	REAL	Rücksetzwert Stellgröße
FEED_FWD	REAL	Störgrößeneingang
OFF	REAL	Offset für P-/PD-Betrieb
Y	REAL	Stellgröße
ERR	REAL	Regeldifferenz
STATUS	Stat_COMP_PID	Ausgangs-Status
SP_CAS_N	REAL	Führungsgröße für Kaskade
YMAN_N	REAL	Hand-Stellwert
OFF_N	REAL	Offset für P-/PD-Betrieb

Parameterbeschreibung Mode_ COMP_PID

Beschreibung der Datenstuktur

Element	Datentyp	Bedeutung
r	BOOL	"1": Betriebsart Rücksetzen
man	BOOL	"1": Betriebsart Hand
halt	BOOL	"1": Betriebsart Halt
cascade	BOOL	"1": Kaskadierungs-Modus
en_p	BOOL	"1": P-Anteil ein
en_i	BOOL	"1": I-Anteil ein
en_d	BOOL	"1": D-Anteil
d_on_pv	BOOL	"1": D-Anteil auf Regelgröße; "0": D-Anteil auf Regeldifferenz
halt_aw	BOOL	"1": Antiwindup Halt "0": Antiwindup Reset
bump	BOOL	"0": Stoßfreier Betriebsartenwechsel
ymanc	BOOL	"1": YMAN nachführen

Parameterbeschreibung Para_COMP_PID

Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
gain	REAL	Proportionalbeiwert (Verstärkungsfaktor)
ti	TIME	Nachstellzeit
td	TIME	Vorhaltezeit
td_lag	TIME	Verzögerungszeit des D-Anteils
db	REAL	Unempfindlichkeitszone
gain_red	REAL	Verstärkungsreduktion in der Unempfindlichkeitszone (db)
rate_sp	REAL	Anstiegsgeschwindigkeit des Sollwerts (SP) [1/s]
rate_man	REAL	Anstiegsgeschwindigkeit des Handstellwerts (YMAN) [1/s]
ymax	REAL	Oberer Grenzwert für Y
ymin	REAL	Unterer Grenzwert für Y
delt_aw	REAL	Begrenzungserweiterung für Antiwindup

Parameterbeschreibung Stat_COMP_PID

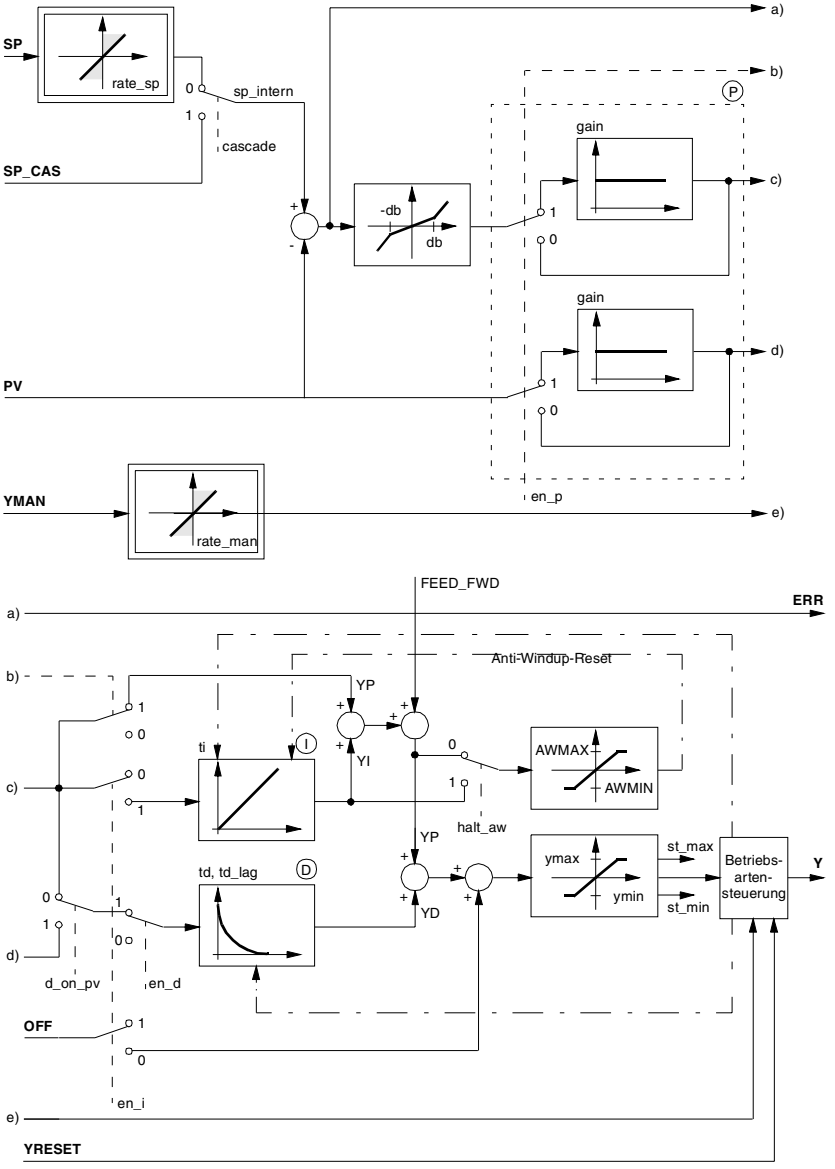
Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
st_r	BOOL	"1": COMP_PID ist in der Betriebsart Rücksetzen
st_man	BOOL	"1": COMP_PID ist in der Betriebsart Hand
st_halt	BOOL	"1": COMP_PID ist in der Betriebsart Halt
st_auto	BOOL	"1": COMP_PID ist in der Betriebsart Automatik
st_cascade	BOOL	"1": COMP_PID ist in der Betriebsart Kaskade
st_max	BOOL	"1": Y ≥ Para_COMP_PID.ymax
st_min	BOOL	"1": Y ≤ Para_COMP_PID.ymin

Strukturbild des komplexen PID-Reglers

Strukturbild

Nachfolgend finden Sie das Strukturbild des COMP_PID-Reglers:



Parametrierung des COMP_PID-Reglers

Parametrierung

Die Struktur des COMP_PID-Reglers ist im *Strukturbild, S. 80* dargestellt.

Die Parametrierung des Funktionsbausteins erfolgt zunächst durch die reinen PID-Parameter, nämlich den Proportionalbeiwert gain, der Nachstellzeit ti und der Vorhaltezeit td.

Der D-Anteil wird mit der Zeit td_lag verzögert. Das Verhältnis von td/td_lag nennt man Differenzierverstärkung und wird im allgemeinen zwischen 3 und 10 gewählt. Der D-Anteil kann entweder basierend auf der Regeldifferenz ERR (d_on_pv = 0) oder basierend auf der Regelgröße PV (d_on_pv = 1) gebildet werden. Wird der D-Anteil aufgrund der Regelgröße PV bestimmt, so wird bei Führungsgrößenänderungen (Änderungen im Eingang SP) kein Sprung bedingt durch den D-Anteil entstehen. Der D-Anteil wirkt sich im Prinzip nur auf Störungen und Prozeßänderungen aus.

Hinweis: Der EFB verfügt über 3 E/A-Paramter (SP_CAS, OFF, YMAN), die von der Funktion im Kaskadenbetrieb selbst aktualisiert werden. Wenn Sie den Baustein im Kaskadierungsbetrieb verwenden wollen, müssen Sie die Verbindung zwischen diesen Eingängen und den entsprechenden Ausgängen (SP_CAS_N, OFF_N, YMAN_N) über Variablen selbst herstellen.

Umkehr des Regelsinns

Ein entgegengesetztes Verhalten des Reglers kann durch Umkehrung des Vorzeichens an gain erreicht werden. Ein positiver Wert an gain bewirkt, daß bei einer positiven Störgröße die Stellgröße steigt. Ein negativer Wert an gain bewirkt, daß bei einer positiven Störgröße die Stellgröße fällt.

Bildung der Regeldifferenz

In der Betriebsart Kaskade wird die Regeldifferenz ERR durch SP_CAS und PV gebildet:

- sp_intern = SP_CAS
- ERR = sp_intern - PV

In der Betriebsart Automatik wird die Regeldifferenz durch sp_intern und PV gebildet, wobei sp_intern über einen Anstiegsbegrenzer auf den Wert des Parameters SP gefahren wird. Die interne Führungsgröße sp_intern wird mit der im Parameter rate_sp festgelegten Anstiegsgeschwindigkeit (Einheit 1/s) rampenförmig auf den Wert des Parameters SP gefahren.

Vom Parameter rate_sp wird der Betrag ausgewertet. Ist rate_sp = 0, so ist die Funktion des Anstiegsbegrenzers für SP abgeschaltet. SP wird direkt auf sp_intern übertragen.

In den Betriebsarten Reset, Hand und Halt wird die Regeldifferenz abhängig vom Zustand des Parameters cascade bestimmt.

Ist cascade=1, so wird sp_intern auf den Wert des Parameters PV eingestellt und ERR ist dann 0.

Ist cascade=0 und ist stoßfreier Betrieb eingestellt ist (bump=0), so wird sp_intern auf den Wert des Parameters SP eingestellt. Ansonsten (bump=1) wird auch hier sp_intern auf den Wert des Parameters PV eingestellt.

Verstärkungs-reduktion bei kleiner Regeldifferenz

Der Parameter db gibt die Größe einer Unempfindlichkeitszone an, in der nicht der volle Proportionalbeiwert gain wirksam ist, sondern ein mit dem Parameter gain_red reduzierter Proportionalbeiwert. Der Parameter db wirkt sich in der im Bild *Darstellung der Unempfinlichkeitszone, S. 83* dargestellten Form auf die Regeldifferenz ERR = SP - PV aus. Durch die Unempfindlichkeitszone können unnötige Stellgliedbelastungen durch kleine Störungen der Regelgröße bzw. durch Messrauschen verringert werden.

Der Parameter db ist positiv einzugeben.

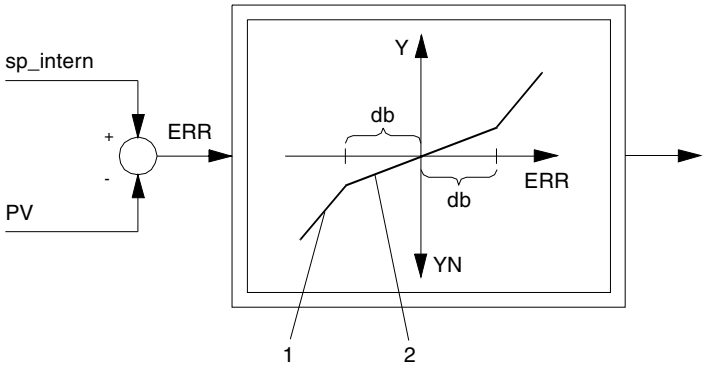
Für gain_red sind Werte zwischen 0 und 1 einzutragen.

Nachführen des Handwertes YMAN

In den Betriebsarten Automatik und Kaskade wird der Eingang YMAN dem Wert der Stellgröße Y nachgeführt, wenn die Betriebsart Handwert-Nachführen eingeschaltet ist (ymanc = 1), d.h. es gilt dann: YMAN = Y. Ist die Betriebsart Handwert-Nachführen ausgeschaltet (ymanc = 0) bleibt der Wert in YMAN unverändert.

Darstellung der Unempfinlich-keitszone

Unempfindlichkeitszone:



1 Steigung 1
2 Steigung gain_red

Stellgrößen-Begrenzung

Die Grenzen ymax und ymin begrenzen die Stellgröße nach oben sowie nach unten. Damit gilt ymin ≤ Y ≤ ymax. .

Das Erreichen der Grenzwerte, bzw. eine Begrenzung der Stellgröße wird durch die beiden Elemente st_max und st_min angezeigt:

- st_max = 1 wenn Y ≥ ymax
- st_min = 1 wenn Y ≤ ymin.

Die Obergrenze ymax für die Begrenzung der Stellgröße ist größer als die Untergrenze ymin zu wählen.

Antiwindup für COMP_PID

Definition

Durch die Antiwindup-Maßnahme wird sichergestellt, daß der Integralanteil nicht zu groß werden kann und der Regler hängt, wenn er zu lange an einer Stellgrenze begrenzt wurde. Die Antiwindup-Maßnahme wird nur bei aktivem I-Anteil des Reglers durchgeführt.

Die Grenzen für die Antiwindup-Maßnahme sind standardmäßig die Stellgrößen des Reglers (delt_aw = 0). Durch den Parameter delt_aw können die Grenzen jedoch gegenüber den Stellgrenzen (ymax, ymin) vergrößert (delt_aw > 0) oder verkleinert (delt_aw < 0) werden.

Die für die Antiwindup-Maßnahme benutzten Grenzen sind demzufolge:

- AWMAX = ymax + delt_aw
- AWMIN = ymin - delt_aw.

Speziell bei stark verrauschten Signalen kann durch die Verschiebung der Antiwindup-Grenzen gegenüber den Stellgrenzen erreicht werden, daß die Stellgröße Y nicht ständig von der Stellgrenze 'wegspringt' (Wirkung des D-Anteils auf Störungen) und anschließend wieder in die Grenzlage hineinfährt (Wirkung des I-Anteils bei Regeldifferenz ERR $\neq$ 0). Sollen die Stellgrenzen gleichzeitig für die Antiwindup-Maßnahme gelten, so ist der Parameter delt_aw = 0 zu wählen.

Durch negative Werte in delt_aw kann dafür gesorgt werden, daß die Grenzen für Antiwindup kleiner sind als die Stellgrenzen (sinnvoll für Antiwindup-Halt).

Antiwindup-Reset (halt_aw = 0)

Der D-Anteil wird für die Antiwindup-Maßnahme nicht mit berücksichtigt, so daß Spitzen, hervorgerufen duch den D-Anteil, nicht durch die Antiwindup-Maßnahme gekappt werden. Die Antiwindup-Reset-Maßnahme korrigiert den I-Anteil in der Form, daß gilt: AWMIN $\leq$ YP + FEED_FWD + YI $\leq$ AWMAX.

Antiwindup-Halt (halt_aw = 1)

Für die Antiwindup-Maßnahme wird nur der I-Anteil berücksichtigt. Wenn Antiwindup-Halt und I-Anteil eingeschaltet ist, korrigiert die Antiwindup-Maßnahme den I-Anteil in der Form, daß gilt: AWMIN $\leq$ YI $\leq$ AWMAX.

Die Parameter rate_sp und rate_man stellen Anstiegsbegrenzer für die Handwerte SP und YMAN dar (siehe dazu auch Funktionsbaustein VLIM). Durch den Wert 0 (rate_sp = 0 bzw. rate_man = 0) wird die Funktion des jeweiligen Anstiegsbegrenzers abgeschaltet. Die Werte SP und YMAN werden dann unverzögert benutzt.

Auswahl des Reglertyps für COMP_PID

Reglertypen

Es gibt verschiedene Reglertypen, die über die Parameter en_p, en_i und en_d ausgewählt werden.

Reglertyp	en_p	en_i	en_d
P-Regler	1	0	0
PI-Regler	1	1	0
PD-Regler	1	0	1
PID-Regler	1	1	1
I-Regler	0	1	0

Der I-Anteil kann auch mit ti = 0 abgeschaltet werden.

Der D-Anteil kann auch mit td = 0 abgeschaltet werden.

Einfluß des Parameters OFF

Ist der I-Anteil eingeschaltet (en_i = 1), so wird die Stellgröße Y aus der Summation der Anteile YP, YI, YD und FEED_FWD ermittelt. Der Offset geht bei eingeschaltetem I-Anteil nicht in die Berechnung mit ein.

Ist der I-Anteil jedoch ausgeschaltet (en_i = 0), so wird die Stellgröße aus der Summation der Anteile YP, YD, FEED_FWD und dem Offset OFF gebildet.

Hinweis: Der Parameter OFF ist nur für P-, D-, oder PD-Regler vorgesehen.

Stoßfreie Betriebsartenumschaltung

Art des Umschaltens

Das Ein- und Aus-Schalten der verschiedenen Anteile (P, I, D) wird stoßfrei realisiert.

Stoßfreies Umschalten bei eingeschaltetem I-Anteil

Wird der P-Anteil zu- bzw. abgeschaltet, so wird der interne I-Anteil um den P-Anteil korrigiert. Auf diese Weise findet eine stoßfrei Zu- bzw. Ab-Schaltung des P-Anteils statt, auch wenn die Regeldifferenz von Null verschieden ist.

Wird der D-Anteil abgeschaltet, so übernimmt der interne I-Anteil den verbleibenden D-Anteil. Wird der D-Anteil zugeschaltet, so wird dessen Anteil auf 0 gesetzt.

Stoßfreies Umschalten bei abgeschaltetem D-Anteil

Stoßfreies Umschalten bei abgeschaltetem D-Anteil wird nur dann realisiert, wenn der Parameter bump = 0 ist. In diesem Fall wird der Parameter OFF dazu benuzt, die stoßfreie Umschaltung zu ermöglichen.

Wird der P-Anteil zu- bzw. abgeschaltet, so wird der Wert im Parameter OFF um den P-Anteil korrigiert. Auf diese Weise findet eine stoßfrei Zu- bzw. Ab-Schaltung des P-Anteils statt, auch wenn die Regeldifferenz von Null verschieden ist.

Wird der D-Anteil abgeschaltet, so wird der verbleibenden D-Anteil auf den Wert des Parameters OFF addiert. Wird der D-Anteil zugeschaltet, so wird dessen Anteil auf 0 gesetzt (OFF bleibt unverändert).

Stoßfreies Umschalten des I-Anteils

Stoßfreies Abschalten des I-Anteils wird nur dann realisiert, wenn der Parameter bump = 0 ist. In diesem Fall wird sowohl der Parameter OFF als auch der interne I-Anteil (YI) dazu benuzt, die stoßfreie Umschaltung zu ermöglichen.

Stoßfreies Umschalten von einem PI(D)- auf P(D)-Regler

Die grundsätzliche Überlegung für eine stoßfreies Umschalten von einem PI(D)- auf P(D)-Regler basiert auf der Annahme, daß der PI(D)-Regler einen statischen Zustand angenommen hat. In diesem Fall ist der Prozeß zur Ruhe gekommen. Der I-Anteil hat in diesem Fall einen spezifischen Wert. Um nun stoßfrei auf P(D)-Betrieb umschalten zu können, müßte der I-Anteil des PI(D)-Reglers als Arbeitspunkt (Offset) für den PD-Regler dienen, so daß die Umschaltung ohne Ausgleichsvorgänge (neues Einschwingen) stattfinden kann. Auf Basis der obigen Überlegung ist das stoßfreie Abschalten des I-Anteils so realisiert, daß der Parameter OFF dessen Wert übernimmt.

Wert der Stellgröße Y in Abhängigkeit von en_i:

Wenn...	Dann...
en_i = 1	Y = YP + YI + YD + FEED_FWD
en_i = 0	Y = YP + OFF + YD + FEED_FWD

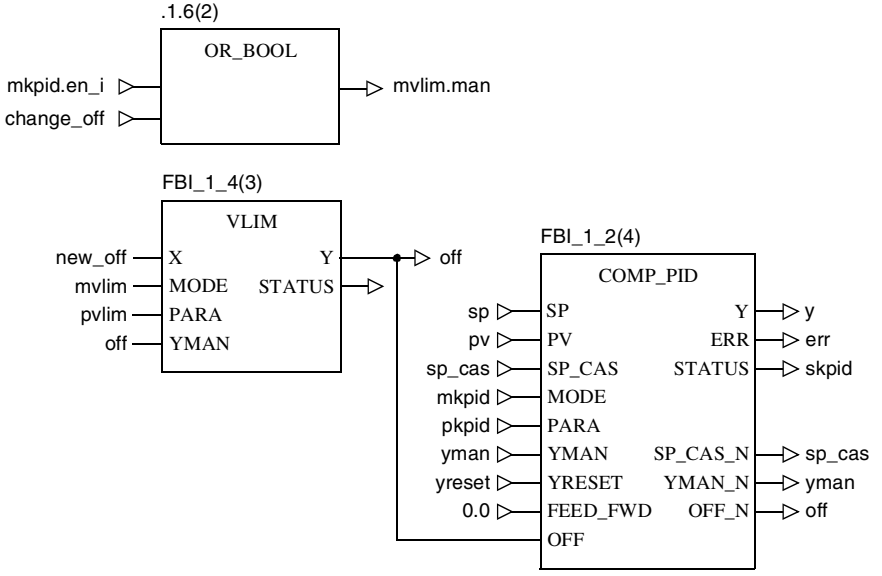
Einschalten des I-Anteils

Das Einschalten des I-Anteils basiert auf einer analogen Überlegung. Der interne I-Anteil wird auf den Wert des Parameters OFF gesetzt. Auf diese Weise kann der I-Anteil zugeschaltet werden ohne daß Ausgleichsvorgänge auftreten.

Hinweis: Wird der Parameter OFF durch einen vorherigen Funktionsbaustein berechnet (Ausgang eines EFBs oder DFBs, z.B. MOVE), so werden die Korrekturen für stoßfreies Umschalten unwirksam (spätestens dann, wenn dieser Funktionsbaustein bearbeitet wird).

Beispiel für stoßfreies Umschalten des D-Anteils

Will man sowohl die stoßfreie Umschaltung des P(D)-Reglers realisieren als auch den Parameter OFF durch das Anwenderprogramm verändern, so kann nachfolgendes Beispiel als Anhaltspunkt dienen.



In diesem Beispiel wird der Parameter OFF über einen Anstiegsbegrenzer VLIM rampenförmig auf den Wert der Variablen new_off gefahren, mit der in pvlim.rate angegebenen Anstiegsgeschwindigkeit.

Hinweise zum Beispiel

Wichtig ist bei diesem Beispiel die Verwendung der Variablen off sowohl am Eingang YMAN des VLIM als auch am Ausgang Y des VLIM und die Verbindung des Ausgangs von VLIM zum Eingang OFF des COMP_PID. Die Verbindung zwischen Ausgang Y von VLIM und Eingang OFF von COMP_PID bewirkt, daß der Funktionsbaustein VLIM vor dem Funktionsbaustein COMP_PID bearbeitet wird (das ist Vorraussetzung für die richtige Funktionsweise). Solange die Betriebsart Hand (mvlim.man = 1) im VLIM eingeschaltet ist, wird der Handwert des Funktionsbausteins VLIM auf den Parameter OFF des COMP_PID übertragen. Der Funktionsbaustein COMP_PID kann nun den Inhalt der Variablen für stoßfreie Behandlung modifizieren. Im nächsten Zyklus steht dieser veränderte Wert ebenso am Eingang YMAN des Funktionsbausteins VLIM zur Verfügung. Zu einem geeigneten Zeitpunkt kann nun die Betriebsart Hand im Funktionsbaustein VLIM ausgeschaltet werden und der Funktionsbaustein fährt den Wert der Variablen off von seinem aktuellen Wert aus rampenförmig auf den Wert von new_off. Im Funktionsbaustein OR_BOOL wird in obigem Beispiel das Einschalten des Handwertes gesteuert. Solange der COMP_PID den I-Anteil eingeschaltet hat (mkpid.en_i = 1) bleibt der Funktionsbaustein VLIM in der Betriebsart Hand.

Hinweis: Ist mkpid.en_i = 1 so geht der Parameter OFF von COMP_PID ohnehin nicht in die Berechnung des Ausgangs von COMP_PID ein.

Im obigen Beispiel wird im Funktionsbaustein OR_BOOL zusätzlich eine zweite Bedingung benötigt, um off auf new_off zu verändern: Die Variable change_off muß 1 sein.

Stoßfreie Änderung der Verstärkung

Das Verändern des Proportionalbeiwertes gain geschieht stoßfrei. Wie bei dem Zu- oder Abschalten von Betriebsarten wird dafür intern eine Korrektur vorgenommen.

Ist der I-Anteil eingeschaltete (en_i = 1 und ti > 0), so wird der interne I-Anteil um den zu erwartenden Sprung im P-Anteil - hervorgerufen durch die Änderung der Verstärkung - korrigiert.

Ist der I-Anteil abgeschaltet, so wird der Wert im Parameter OFF um den zu erwartenden Sprung im P-Anteil korrigiert, wenn der Parameter bump = 0 ist. Ist bump = 1 so wird OFF nicht verändert und eine Verstärkungsänderung des P(D)-Reglers führt zu Ausgleichsvorgängen.

Auswahl der Betriebsart des COMP_PID

Betriebsarten

Es gibt fünf Betriebsarten, die über r, man , halt und cascade ausgewählt werden.

Betriebsart	r	man	halt	cascade
Reset	1	1 oder 0	1 oder 0	1 oder 0
Hand	0	1	1 oder 0	1 oder 0
Halt	0	0	1	1 oder 0
Kaskade	0	0	0	1
Automatik	0	0	0	0

Betriebsart Automatik und Kaskade

In der Betriebsart Automatik wird die Stellgröße Y durch den diskretisierten PID-Algorithmus in Abhängigkeit von der Regelgröße X und der Führungsgröße SP bestimmt.

In der Betriebsart Kaskade wird die Stellgröße Y durch den diskretisierten PID-Algorithmus in Abhängigkeit von der Regelgröße X und der Führungsgröße SP_CAS bestimmt.

Diese beiden Betriebsarten Automatik und Kaskade unterscheiden sich äußerlich nur duch die unterschiedliche Verwendung der Führungsgröße SP. SP_CAS gilt für Kaskade, SP für alle anderen Betriebsarten (mit Anstiegsbegrenzung). Die Größe SP_CAS ist nur in der Betriebsart Kaskade ein Eingang und in allen anderen Betriebsarten ein Ausgang. In den Betriebsarten Reset, Hand, Halt oder Automatik sowie im Anlauf wird in SP_CAS die Größe X an den Führungsregler zurückgegeben, um stoßfreies Umschalten von z.B. Festwertregelung auf Kaskadenregelung zu ermöglichen.

In beiden Betriebsarten wird die Stellgröße Y durch ymax und ymin begrenzt. Für die Antiwindup-Maßnahme können die Stellgrenzen durch den Parameter delt_aw aufgeweitet werden.

Betriebsart Hand

In der Betriebsart Hand wird der Handstellwert YMAN mittels eines Anstiegsbegrenzers auf die Stellgröße Y übertragen. Die Stellgröße Y wird mit der im Parameter rate_man festgelegeten Anstiegsgeschwindigkeit (Einheit 1/s) rampenförmig auf den Wert des Parameters YMAN gefahren.

Vom Parameter rate_man wird der Betrag ausgewertet. Ist rate_man = 0, so ist die Funktion des Anstiegsbegrenzers für YMAN abgeschaltet. YMAN wird direkt auf die Stellgröße übertragen. Die Stellgröße wird durch ymax und ymin begrenzt.

Die internen Größen werden so nachgeführt, daß der Regler (bei zugeschaltetem I-Anteil) stoßfrei von Hand auf Automatik umgeschaltet werden kann. Die Antiwindup-Maßnahme ist wie im Automatik-Betrieb ausgelegt.

Der D-Anteil wird in dieser Betriebsart automatisch auf 0 gesetzt.

Betriebsart Reset

In der Betriebsart Reset wird der Reset-Wert YRESET fest auf die Stellgröße Y übertragen. Die Stellgröße wird durch ymax und ymin begrenzt. Die internen Größen werden so nachgeführt, daß der Regler (bei zugeschaltetem I-Anteil) stoßfrei von Hand auf Automatik umgeschaltet werden kann. Die Antiwindup-Maßnahme ist wie im Automatik-Betrieb realisiert.

Betriebsart Halt

In der Betriebsart Halt bleibt die Stellgröße, wie sie vorgefunden wird, d.h. der Funktionsbaustein verändert die Stellgröße Y nicht (Regler bleibt stehen). Die internen Größen werden so nachgeführt, daß der Regler ohne Stoß von seiner aktuellen Lage aus weiterfährt. Stellgrößenbegrenzung und Antiwindup-Maßnahme sind wie im Automatik-Betrieb. Die Betriebsart Halt eignet sich unter anderem auch dafür, die Stellgröße Y über ein externes Bediengerät zu stellen, wobei die internen Anteile im Regler richtig nachgeführt werden.

Der D-Anteil wird in dieser Betriebsart automatisch auf 0 gesetzt.

Nicht-Stoßfreier-Betrieb (bump = 0)

Nicht-stoßfreier Betrieb wird so definiert, daß der Regler bei Betriebsartenumschaltung (z.B. Hand-Automatik) einen Sprung bedingt durch den P-Anteil in der Stellgröße Y aufweist. Abhängig vom Einsatzgebiet des Reglers kann es sinnvoll sein, daß der Regler bei Umschaltung z.B. von Hand auf Automatik die Stellgröße sprungartig korrigiert, wenn die Regeldifferenz ungleich 0 ist.

Die Höhe dieses Sprungs entspricht dem P-Anteil des Reglers und beträgt:

YP = ERR x gain

Stoßfreier Betrieb (bump = 1)

Stoßfreier Betrieb wird so definiert, daß der Regler bei Betriebsartenumschaltung im Stellgrößenverlauf Y keine Unstetigkeitsstelle aufweist. Das heißt, er soll ohne Sprung an genau derselben Stelle weiterfahren, wo er zuletzt stand. Der interne I-Anteil wird in dieser Betriebsart um den P-Anteil mit korrigiert. Ist kein I-Anteil eingeschaltet wird der stoßfreie Betrieb dadurch erreicht, daß der Arbeitspunkt OFF so nachgefahren wird, daß der Regler trotz Regeldifferenz ungleich 0 ohne Stoß bei Betriebsartenwechsel weiterfahren kann.

Detaillierte Formeln

Erläuterung der Formelgrößen

Bedeutung der Größen in den nachfolgenden Formeln:

Größe	Bedeutung
dt	Zeitdifferenz zwischen aktullem Zyklus und vorherigem Zyklus
ERR	Die aktuell intern gebildete Regeldifferenz
$ERR_{(new)}$	Wert der Regeldifferenz aus dem aktuellen Abtastschritt
$ERR_{(old)}$	Wert der Regeldifferenz aus dem vorherigen Abtastschritt
FEED_FWD	Störgröße (nur bei P-, D- oder PD-Regler
OFF	Offset
$PV_{(new)}$	Wert der Regelgröße aus dem aktuellen Abtastschritt
$PV_{(old)}$	Wert der Regelgröße aus dem vorherigen Abtastschritt.
Y	aktueller Ausgang (Betriebsart Halt) oder YMAN (Betriebsart Hand)
YD	D-Anteil (nur bei en_d=1)
YI	I-Anteil (nur bei en_i=1)
YP	P-Anteil (nur bei en_p=1)

Stellgröße

Die Stellgröße setzt sich aus verschiedenen von den Betriebsarten abhängigen Teilgrößen zusammen:

$$Y = YP + YI + YD + OFF + FEED_FWD$$

Nach der Summation der Anteile findet eine Stellgrößen-Begrenzung statt, so daß gilt:

$$ymin \leq Y \leq ymax$$

Übersicht zur Berechnung der Regler-Anteile

Nachfolgend finden Sie eine Übersicht über die unterschiedliche Berechnung der Regler-Anteile in Abhängigkeit von den Elementen en_p, en_i und en_d:

- P-Anteil YP für Betriebsart Hand, Halt, Automatik und Kaskade
- I-Anteil YI für Betriebsart Automatik
- I-Anteil YI für Betriebsart Hand und Halt
- D-Anteil YD für Betriebsart Automatik und Kaskade
- D-Anteil YD für Betriebsart Hand und Halt

P-Anteil YP für alle Betriebsarten

YP für Betriebsart Hand, Halt, Automatik und Kaskade wird wie folgt ermittelt:

Für en_p = 1 gilt:

$$YP = gain \times ERR$$

Für en_p = 0 gilt:

$$YP = 0$$

I-Anteil YI für Betriebsart Automatik

YI für Betriebsart Automatik wird wie folgt ermittelt:

Für en_i = 1 gilt:

$$YI_{(new)} = YI_{(old)} + gain \times \frac{dt}{ti} \times \frac{ERR_{(new)} + ERR_{(old)}}{2}$$

Für en_i = 0 gilt:

$$YI = 0$$

Der I-Anteil wird nach der Trapezregel gebildet.

I-Anteil YI für Betriebsart Hand und Halt

YI für Betriebsart Hand und Halt wird wie folgt ermittelt:

Für en_i = 1 gilt:

$$YI = Y - (YP - FEED_FWD)$$

Für en_i = 0 gilt:

$$YI = 0$$

D-Anteil YD für Betriebsart Automatik und Kaskade

YD für Betriebsart Automatik und Kaskade wird wie folgt ermittelt:

Für en_d = 1 und d_on_pv = 0 gilt:

$$YD_{(new)} = \frac{YD_{(old)} \times td_lag + td \times gain \times (ERR_{(new)} - ERR_{(old)})}{dt + dt_lag}$$

Für en_d = 1 und d_on_pv = 1 gilt:

$$YD_{(new)} = \frac{YD_{(old)} \times td_lag + td \times gain \times (PV_{(old)} - PV_{(new)})}{dt + dt_lag}$$

Für en_d = 0 gilt:

$$YD = 0$$

D-Anteil YD für Betriebsart Hand und Halt

YD für Betriebsart Hand und Halt wird wie folgt ermittelt:

$$YD = 0$$

Laufzeitfehler

Fehlermeldung

Es erfolgt eine Fehlermeldung, wenn

- eine unzulässige Gleitkommazahl am Eingang PV anliegt
- gain_red > 1 oder gain_red < 0 ist
- db < 0 ist
- ymax < ymin ist

DEADTIME: Totzeitglied

7

Übersicht

Einleitung

Dieses Kapitel beschreibt den Baustein DEADTIME.

Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Kurzbeschreibung	96
Darstellung	97
Betriebsarten	98
Beispiel zum Verhalten des Funktionsbausteines	99
Laufzeitfehler	99

Kurzbeschreibung

Funktionsbeschreibung

Mit diesem Funktionsbaustein wird ein Eingangssignal um eine Zeit verzögert, der sogenannten Totzeit.

Der Funktionsbaustein verzögert das Signal X um die Totzeit T_DELAY, bevor es an Y wieder ausgegeben wird.

Der Funktionsbaustein hat einen Puffer für 128 zeitlich aufeinanderfolgende X-Werte, d.h. während der Zeit T_DELAY werden 128 X-Werte gespeichert. Der Puffer wird entsprechend den verschiedenen Betriebsarten behandelt.

Nach einem Kalt- oder Warmstart des Systems bleibt der Wert von Y unverändert. Die internen Werte werden auf den Wert von X gesetzt.

Nach einem Kalt- oder Warmstart des Systems oder einer Änderung der Totzeit T_DELAY, wird der Ausgang READY gleich "0". Dies bedeutet: nicht bereit, leerer Puffer.

Der Funktionsbaustein hat die Betriebsarten:
- Hand
- Halt
- Automatik.

Als zusätzliche Parameter können EN und ENO projektiert werden.

Hinweis: Die Verzögerungszeit läuft weiter, auch wenn der Baustein über den Parameter EN gesperrt wird, da der Baustein seine Zeitdifferenzen nach der Systemuhr berechnet.

Formel

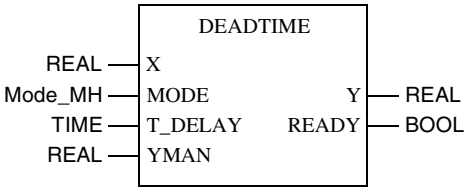
Die Übertragungsfunktion lautet:

$$G(s) = e^{-s \times T_DELAY}$$

Darstellung

Symbol

Darstellung des Bausteins



Parameterbeschreibung DEADTIME

Beschreibung der Bausteinparameter

Parameter	Datentyp	Bedeutung
X	REAL	Eingangswert
MODE	Mode_MH	Betriebsart
T_DELAY	TIME	Totzeit
YMAN	REAL	HandStellwert
Y	REAL	Ausgang
READY	BOOL	"1" = Interner Puffer gefüllt "0" = Interner Puffer nicht gefüllt (z.B. nach Warm/ Kaltstart oder Änderung der Totzeit)

Parameterbeschreibung Mode_MH

Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
man	BOOL	"1" = Betriebsart Hand
halt	BOOL	"1" = Betriebsart Halt

Betriebsarten

Auswahl der Betriebsarten

Es gibt drei Betriebsarten, die über die Elemente man und halt ausgewählt werden:

Betriebsart	man	halt
Automatik	0	0
Hand	1	0 oder 1
Halt	0	1

Betriebsart Automatik

In der Betriebsart Automatik arbeitet der Funktionsbaustein nach folgenden Regeln:

Wenn...	Dann...
Zykluszeit > $\frac{T_Delay}{128}$	wird der aktuelle X-Wert in den Puffer übertragen, und der älteste X-Wert im Puffer wird am Ausgang Y ausgegeben. Bei einer Zykluszeit größer als T_DELAY / 128, ergibt sich eine Auflösung kleiner 128 und es kommt zu einem systematischem Fehler, d.h. einige X-Werte werden doppelt gespeichert (siehe nachfolgendes Beispiel).
Zykluszeit < $\frac{T_Delay}{128}$	können nicht alle X-Werte im Puffer abgelegt werden. In diesem Fall wird in einigen Zyklen der X-Wert nicht gespeichert. Entsprechend bleibt der Ausgang Y nach Ablauf der Zeit T_DELAY in zwei (oder mehr) aufeinanderfolgenden Zyklen unverändert.

Beispiel zur Betriebsart Automatik

Im Beispiel werden folgende Werte angenommen:

Zykluszeit = 100 ms

T_DELAY = 10 s

tin = T_DELAY / 128 = 78 ms

Da die Einlesezeit tin kleiner ist als die Zykluszeit, wird jeder X-Wert in den Puffer übernommen. Bei der vierten Ausführung des Funktionsbausteins (nach 400 ms) wird der X-Wert jedoch zweimal anstatt einmal gespeichert (weil 3 x 78 = 312 und 4 x 78 = 390).

Betriebsart Hand

In der Betriebsart Hand wird der Handwert YMAN fest auf den Ausgang Y übertragen. Der interne Puffer wird mit dem Handwert YMAN gefüllt. Der Puffer wird als gefüllt markiert (READY = 1).

Betriebsart Halt

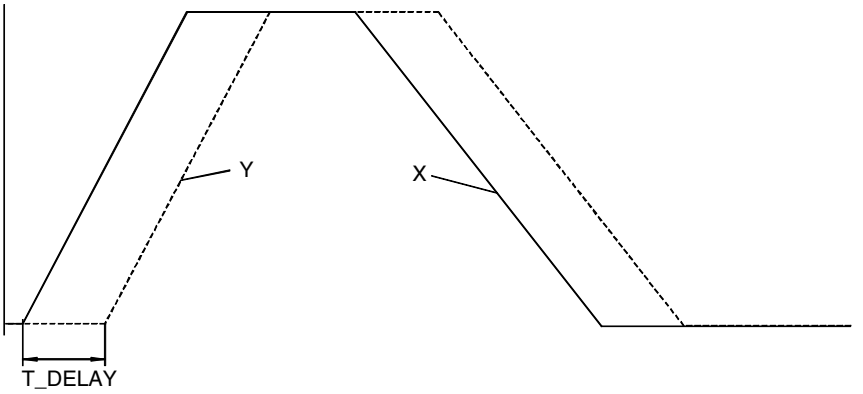
In der Betriebsart Halt wird der Ausgang Y auf dem zuletzt berechneten Wert gehalten. Der Ausgang wird nicht mehr verändert, kann jedoch vom Anwender überschrieben werden. Der interne Puffer wird jedoch weiter bearbeitet wie in der Betriebsart Automatik.

Beispiel zum Verhalten des Funktionsbausteines

Beispiel

Das folgende Diagramm zeigt ein Beispiel für das Verhalten des Funktionsbausteins. Der Eingang X ändert sich in Form einer Rampe von einem Wert auf einen neuen Wert und der Ausgang Y folgt um die Totzeit T_DELAY verzögert dem Eingang X.

Diagramm Funktionsbaustein DEADTIME



Laufzeitfehler

Fehlermeldung

Es erfolgt eine Fehlermeldung, wenn eine unzulässige Gleitkommazahl am Eingang YMAN oder X anliegt.

DELAY: Totzeitglied

8

Übersicht

Einleitung

Dieses Kapitel beschreibt den Baustein DELAY.

Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Kurzbeschreibung	102
Darstellung	103
Betriebsarten	104
Beispiel zum Verhalten des Funktionsbausteines	105

Kurzbeschreibung

Funktionsbeschreibung

Mit diesem Funktionsbaustein wird ein Eingangssignal um eine Totzeit verzögert.

Der Funktionsbaustein verzögert das Signal X um die Totzeit T_DELAY, bevor es an Y wieder ausgegeben wird.

Der Funktionsbaustein hat einen Delay-Puffer für 128 Elemente (X-Werte), d.h. während der Zeit T_DELAY können 128 X-Werte gespeichtert werden. Der Puffer wird entsprechend den verschiedenen Betriebsarten behandelt.

Nach einem Kalt- oder Warmstart des Systems bleibt der Wert von Y unverändert. Die internen Werte werden auf den Wert von X gesetzt.

Nach einem Kalt- oder Warmstart des Systems oder einer Änderung der Totzeit T_DELAY, wird der Ausgang READY gleich "0". Dies bedeutet: nicht bereit da leerer Puffer.

Der Funktionsbaustein hat die Betriebsarten: Hand, Halt und Automatik.

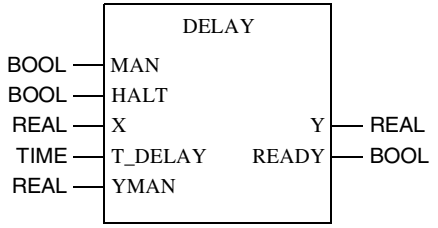
Als zusätzliche Parameter können EN und ENO projektiert werden.

Hinweis: Die Verzögerungszeit läuft weiter, auch wenn der Baustein über den Parameter EN gesperrt wird, da der Baustein seine Zeitdifferenzen nach der Systemuhr berechnet.

Darstellung

Symbol

Darstellung des Bausteins



Parameterbeschreibung

Beschreibung der Bausteinparameter

Parameter	Datentyp	Bedeutung
MAN	BOOL	"1" = Betriebsart Hand
HALT	BOOL	"1" = Betriebsart Halt
X	REAL	Eingangswert
T_DELAY	TIME	Totzeit
YMAN	REAL	HandStellwert
Y	REAL	Ausgang
READY	BOOL	"1" = Interner Puffer gefüllt "0" = Interner Puffer nicht gefüllt (z.B. nach Warm/ Kaltstart oder Änderung der Totzeit

Betriebsarten

Auswahl der Betriebsarten

Es gibt drei Betriebsarten, die über die Eingänge MAN und HALT ausgewählt werden.

Betriebsart	MAN	HALT
Automatik	0	0
Hand	1	0 oder 1
Halt	0	1

Betriebsart Automatik

In der Betriebsart Automatik arbeitet der Funktionsbaustein nach folgenden Regeln:

Wenn...	Dann...
Zykluszeit > $\frac{T_Delay}{128}$	wird der aktuelle X-Wert in den Puffer übertragen, und der älteste X-Wert im Puffer wird am Ausgang Y ausgegeben. Bei einer Zykluszeit größer als T_DELAY / 128, ergibt sich eine Auflösung kleiner 128 und es kommt zu einem systematischem Fehler, d.h. einige X-Werte werden doppel gespeichert. (siehe nachfolgendes Beispiel).
Zykluszeit < $\frac{T_Delay}{128}$	können nicht alle X-Werte im Puffer abgelegt werden. In diesem Fall wird in einigen Zyklen der X-Wert nicht gespeichert, und Y bleibt in diesen Zyklen unverändert

Beispiel zur Betriebsart Automatik

Im Beispiel werden folgende Werte angenommen:

Zykluszeit = 100 ms

T_DELAY = 10 s

tin = T_DELAY / 128 = 78 ms

Da die Einlesezeit tin kleiner ist als die Zykluszeit, wird jeder X-Wert in den Puffer übernommen. Bei der vierten Ausführung des Funktionsbausteins (nach 400 ms) wird der X-Wert jedoch zweimal anstatt einmal gespeichert (weil 3 x 78 = 312 und 4 x 78 = 390).

Betriebsart Hand

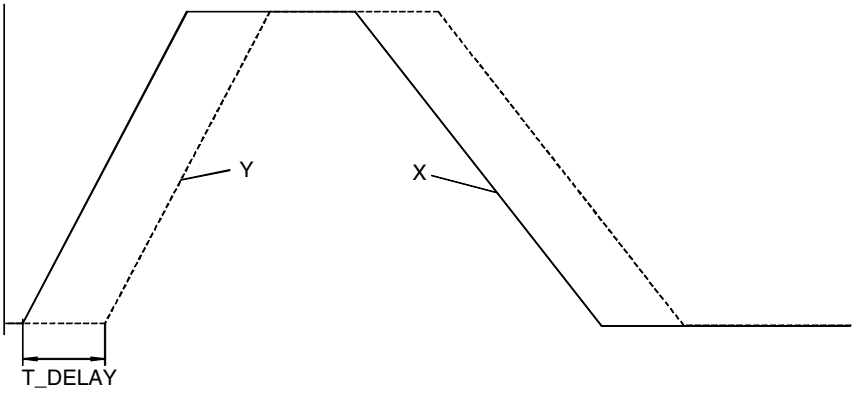
In der Betriebsart Hand wird der Handwert YMAN fest auf den Ausgang Y übertragen. Der interne Puffer wird mit dem Handwert YMAN gefüllt. Der Puffer wird als gefüllt markiert (READY = 1).

Betriebsart Halt In der Betriebsart Halt wird der Ausgang Y auf dem zuletzt berechneten Wert gehalten. Der Ausgang wird nicht mehr verändert, kann jedoch vom Anwender überschrieben werden. Der interne Puffer wird jedoch weiter bearbeitet wie in der Betriebsart Automatik.

Beispiel zum Verhalten des Funktionsbausteines

Beispiel Das folgende Diagramm zeigt ein Beispiel für das Verhalten des Funktionsbausteins. Der Eingang X ändert sich in Form einer Rampe von einem Wert auf einen neuen Wert und der Ausgang Y folgt um die Totzeit T_DELAY verzögert dem Eingang X.

Diagramm Funktionsbaustein DELAY



DERIV: Differenzierer mit Glättung

9

Übersicht

Einleitung

Dieses Kapitel beschreibt den Baustein DERIV.

Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Kurzbeschreibung	108
Darstellung	108
Formeln	109
Detailbeschreibung	110
Beispiel zum Funktionsbaustein	111
Laufzeitfehler	111

Kurzbeschreibung

Funktionsbeschreibung

Der Funktionsbaustein stellt ein Differenzierglied mit einem mit der Zeitkonstante lag verzögerten Ausgang Y dar.

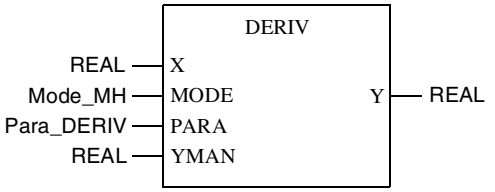
Der Funktionsbaustein besitzt folgende Betriebsarten: Hand, Halt und Automatik.

Als zusätzliche Parameter können EN und ENO projektiert werden.

Darstellung

Symbol

Darstellung des Bausteins



Parameterbeschreibung DERIV

Beschreibung der Bausteinparameter

Parameter	Datentyp	Bedeutung
X	REAL	Eingangsgröße
MODE	Mode_MH	Betriebsarten
PARA	Para_DERIV	Parameter
YMAN	REAL	Hand-Stellwert
Y	REAL	Ausgang Differenzierer mit Glättung

Parameterbeschreibung Mode_MH

Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
man	BOOL	"1" = Betriebsart Hand
halt	BOOL	"1" = Betriebsart Halt

Parameterbeschreibung Para_DERIV

Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
gain	REAL	Verstärkung der Differenzierung
lag	TIME	Verzögerungszeitkonstante

Formeln

Übertragungsfunktion

Die Übertragungsfunktion für Y lautet:

$$G(s) = gain \times \frac{s \times lag}{1 + s \times lag}$$

Berechnungsformel für Y

Die Berechnungsformel für Y lautet:

$$Y = \frac{lag}{dt + lag} \times (Y_{(old)} + gain \times (X_{(new)} - X_{(old)}))$$

Sonderfall: lag =0

Es handelt sich hierbei um die reine Differenzierung ohne Verzögerungsglied 1. Ordnung.

In diesem Fall ist die Übertragungsfunktion:

$$G(s) = gain \times s$$

Die Berechnungsformel lautet:

$$Y = gain \times \frac{X_{(new)} - X_{(old)}}{dt}$$

Bedeutung der Größen

Nachfolgend finden Sie die Bedeutung der Formelgrößen:

Größe	Bedeutung
$X_{(new)}$	der Wert des Eingangs X aus dem aktuellen Zyklus
$X_{(old)}$	der Wert des Eingangs X aus dem vorherigen Zyklus
$Y_{(old)}$	der Wert des Ausgangs Y aus dem vorherigen Zyklus
dt	Zeitdifferenz zwischen aktuellem Zyklus und vorherigem Zyklus

Detailbeschreibung

Parametrierung

Die Parametrierung des Funktionsbausteins erfolgt durch Festlegen der Verstärkung gain des Differenzierers und der Zeitkonstante lag, mit welcher der Ausgang Y verzögert wird.

Für sehr kleine Abtastzeiten und dem Einheitssprung am Eingang X (Sprung am Eingang X von 0 auf 1.0) wird der Ausgang Y auf den Wert gain springen (theoretischer Wert - real ein bißchen kleiner, bedingt durch die nicht unendlich kleine Abtastzeit), um dann mit der Zeitkonstante lag wieder auf 0 zurückzugehen.

Betriebsarten

Es gibt drei Betriebsarten, die über die Elemente man und halt ausgewählt werden:

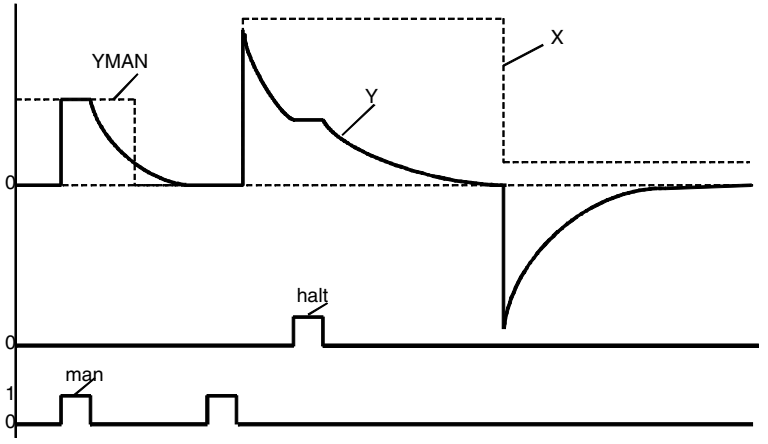
Betriebsart	man	halt	Bedeutung
Automatik	0	0	Der Funktionsbaustein wird, wie in "Parametrierung" beschrieben, bearbeitet.
Hand	1	0 oder 1	Der Eingang YMAN wird auf den Ausgang Y übertragen.
Halt	0	1	Der Ausgang Y wird auf dem zuletzt berechneten Wert gehalten. Der Ausgang wird nicht mehr verändert, kann jedoch vom Anwender überschrieben werden.

Beispiel zum Funktionsbaustein

Beispiel DERIV

Im nachfolgenden Beispiel ist die Sprungantwort des DERIV-Funktionsbausteins dargestellt.

Sprungantwort mit gain = 1 und lag = 10 s



Laufzeitfehler

Fehlermeldung

Es erfolgt eine Fehlermeldung, wenn eine unzulässige Gleitkommazahl am Eingang YMAN oder X anliegt.

DTIME: Verzögerung

10

Übersicht

Einleitung

Dieses Kapitel beschreibt den Baustein DTIME.

Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

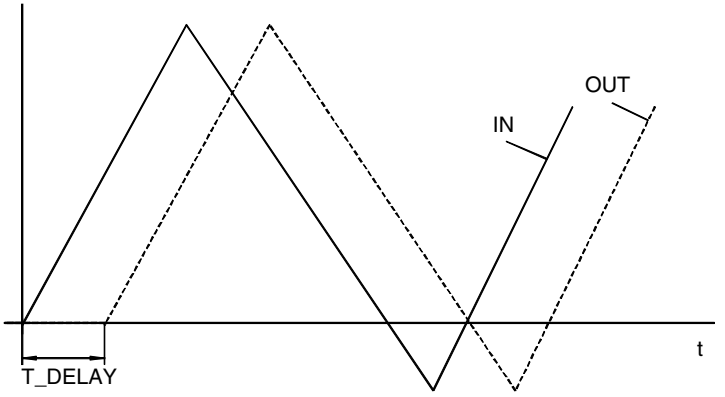
Thema	Seite
Kurzbeschreibung	114
Darstellung	115
Parametrierung	116
Initialisierungen und Betriebsarten	119
Beispiel zum Messen einer Durchflußmenge	120
Laufzeitfehler	121

Kurzbeschreibung

Funktionsbeschreibung

Der FunktionsbausteinDTIME generiert eine Verzögerung bei der Übertragung der numerischen Eingangsgröße IN. Die numerische Ausgangsgröße OUT generiert dasselbe Verhalten wie die numerische Eingangsgröße unter Einbeziehung der Verzögerung T_DELAY, die variieren kann.

Verhalten des Funktionbausteins DTIME:



Als zusätzliche Parameter können EN und ENO projektiert werden.

Formel

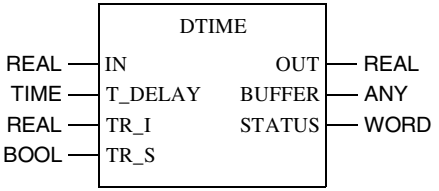
Dieser Funktionsbaustein implementiert folgende Übertragungsfunktion :

$$G_{(p)} = e^{-p.T_DELAY}$$

Darstellung

Symbol

Darstellung des Bausteins



Parameterbeschreibung

Beschreibung der Bausteinparameter

Parameter	Datentyp	Bedeutung
IN	REAL	Zu verzögernder Digitalwert
T_DELAY	TIME	Gewünschte Verzögerung
TR_I	REAL	Initialisierungseingang
TR_S	BOOL	Initialisierungsbefehl
OUT	REAL	Verzögerter Ausgang
BUFFER	ANY*)	Speicher, in dem die verzögerten Werte abgelegt werden.
STATUS	WORD	Statuswort

*) Dieser Parameter muß grundsätzlich mit einer Variablen verknüpft sein (siehe "*Parametrierung, S. 116*").

Parametrierung

Speicherung der Eingangswerte (BUFFER-Ausgang)

Der BUFFER-Ausgang muß mit einer Variablen verknüpft werden (im allgemeinen des Typs Buffer_DTIME). Die zu verzögernden Werte werden in dieser Variablen abgelegt. Jedesmal, wenn der Funktionsblock ausgeführt wird, wird für den Eingang IN ein neuer Wert gespeichert.

Die Größe der mit dem BUFFER-Ausgang verknüpften Variable bestimmt die Anzahl der Werte, die gespeichert werden können und somit ebenfalls den Maximalwert, den die Verzögerung annehmen darf :

$$T_DELAY_{maximum} = n \times T_Period$$

Hierbei gilt

Formelgröße	Bedeutung
n	Anzahl Gleitkommawerte, die in BUFFER enthalten sein kann
T_PERIOD	Abtastintervall des Funktionsbausteins

Hinweis: Sobald eine Variable an den BUFFER-Ausgang angeschlossen wurde, kann sie nur durch eine Variable desselben Typs ersetzt werden. Um diese durch eine größere Variable zu ersetzen, mit der beispielsweise höhere Verzögerungswerte erreicht werden können, muß der Funktionsbaustein gelöscht und ein neuer angelegt werden.

Datentyp des Buffer-Ausgangs

Der BUFFER-Ausgang ist vom Typ ANY. Das heißt, daß ihm ein beliebiger Variablentyp zugeordnet werden kann. Im Prinzip ist es von Vorteil, zunächst einmal eine Variable des Typs Buffer_DTIME zu verwenden. Hierbei handelt es sich um eine Tabelle, die bis zu 100 Gleitwerte enthalten kann. Mit diesem Variablentyp ist es möglich, eine Verzögerung zu erreichen, die dem 100-fachen des Abtastintervalls des Funktionsbausteins DTIME entspricht.

Vorgehensweise bei großen Verzögerungs-zeiten

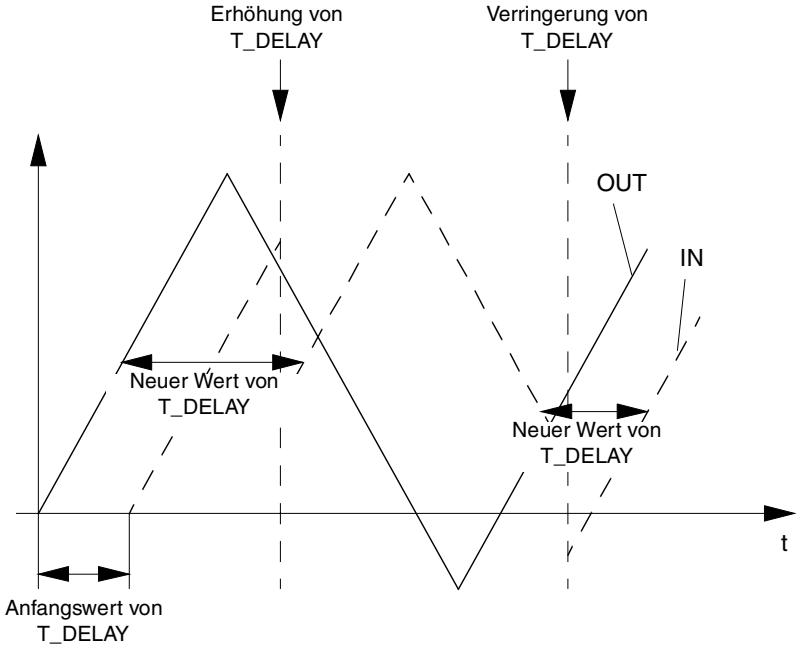
Um Verzögerungswerte zu erreichen, die mehr als dem 100-fachen des Abtastintervalls des Funktionsbausteins entsprechen, muß dem Parameter BUFFER eine größere Variable zugeordnet werden:

Schritt	Aktion
1	Definieren Sie einen neuen abgeleiteten Datentyp, z.B. eine Tabelle mit 200 Gleitkommawerten
2	Deklarieren Sie eine Variable dieses Typs und verknüpfen Sie diese mit dem Parameter BUFFER des Funktionsbausteins DTIME
3	Die maximale Verzögerung entspricht in diesem Fall dem 200-fachen des Abtastintervalls des Funktionsbausteins

Dynamische Änderung der Verzögerung T_DELAY

Es ist möglich, die Verzögerungszeit T_DELAY bei Programmablauf zu erhöhen oder herabzusetzen. Sofern die neu eingestellte Verzögerungszeit mit der Größe des BUFFER-Ausgangs kompatibel ist, ist die neue Verzögerung sofort wirksam.

Darstellung der dynamischen Änderung der Verzögerung T_DELAY



Wenn der Wert von T_DELAY im Verhältnis zur BUFFER-Größe zu groß wird, ist es nicht mehr möglich, genügend Eingangswerte zu speichern, um die gewünschte Verzögerung zu erreichen. Die Verzögerung verbleibt in diesem Fall auf der größtmöglichen Zeitdauer (Bit 8 des Statuswortes geht dann auf 1 über).

Um dieses Problem zu verhindern, ist es empfehlenswert, die dem BUFFER-Parameter zugeordnete Variable so zu dimensionieren, daß mit einer eventuellen Erhöhung der Verzögerung T_DELAY gerechnet werden kann.

Wenn T_DELAY = 0, entspricht der Ausgang OUT immer dem Eingang IN.

Initialisierungen und Betriebsarten

Initialisierungen und Betriebsarten

Bei der ersten Ausführung des Funktionsbausteins (bei Laden des Programms oder Online- Aufruf) werden alle in BUFFER enthaltenen Werte mit dem Wert von TR_I initialisiert. Der Ausgang OUT behält für die Dauer der Verzögerung T_DELAY diesen Wert bei. Wenn der Eingang TR_I nicht angeschlossen ist, dient der Wert 0 zur Initialisierung des BUFFER-Ausgangs, der Ausgang OUT behält während der Verzögerung T_DELAY den Wert 0 bei.

In der Betriebsart Tracking (TR_S = 1) wird der Eingang TR_I auf den Ausgang OUT geführt und der Ausgang BUFFER ebenfalls mit dem Wert von TR_I initialisiert. Nach Rückkehr zur Betriebsart Normal behält der Ausgang für die Dauer von T_DELAY diesen Wert bei, wie dies bereits beim ersten Zyklus der Fall war.

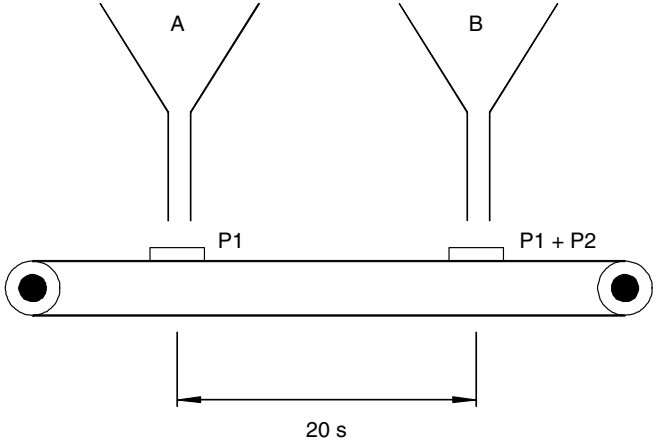
Beispiel zum Messen einer Durchflußmenge

Messen einer Durchflußmenge

Der Funktionsbaustein DTIME dient beispielsweise zur Modellierung einer Prozeßverzögerung, die u.a beim Messen von Durchflußmengen oder Drehzahlen von Antriebssystemen projektiert werden kann.

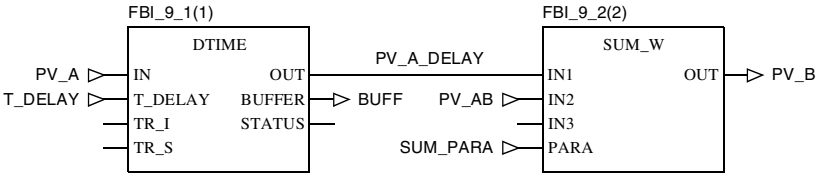
Im nachfolgenden Beispiel werden zwei Produkte, A und B nacheinander in einen Behälter gegossen und gemischt. Der Behälter wird zunächst unter die Dosiereinrichtung für das Produkt A gestellt, die die Menge P1 ausgibt. Er wird dann über ein Fließband zur Dosiereinrichtung für das Produkt B transportiert, die die Menge P2 ausgibt. Das Zeitintervall zwischen den beiden Dosiereinrichtungen beträgt 20 s.

Messen von Durchflußmengen



Die Produktmenge P2 wird geregelt, die Gewichtsmessung am Behälter ergibt aber P1+P2. P1 ist abzuziehen. Die Menge P2 entspricht der gemessenen Menge abzüglich der 20 s zuvor dosierten Menge P1.

Die Messung des Regelkreises an P2 entspricht demnach folgendem Schema:



Werte der Datenstruktur-Elemente der Variablen SUM_PARA:

Element von SUM_PARA	Wert
SUM_PARA.K1	1
SUM_PARA.K2	1

Laufzeitfehler

Statuswort

Im Statuswort werden folgende Meldungen angezeigt:

Bit	Bedeutung
Bit 0 = 1	Fehler in einer Berechnung mit Gleitkommawerten
Bit 1 = 1	Erfassung eines unzulässigen Wertes an einem der Gleitkommaeingänge
Bit 2 = 1	Division durch Null bei einer Berechnung in Gleitkommawerten
Bit 3 = 1	Kapazitätsüberlauf bei einer Berechnung in Gleitkommawerten
Bit 8 = 1	T_DELAY übersteigt den Maximalwert, der über den BUFFER-Ausgang erreicht werden kann

Fehlermeldung

Dieser Fehler erscheint, wenn an einem Eingang ein nicht gleitender Wert erfaßt wird oder wenn bei einer Berechnung in Gleitkommawerten ein Problem auftritt. In diesem Fall bleiben die Ausgänge OUT und BUFFER unverändert.

Alarm

Es wird ein Alarm ausgegeben, wenn T_DELAY den möglichen Maximalwert überschreitet. In diesem Fall verwendet der Funktionsbaustein den Maximalwert. Wenn ein über den Verzögerungswert hinausgehender Wert gewünscht wird, braucht nur der BUFFER-Ausgang mit einer größeren Variablen verknüpft zu werden.

FGEN: Funktionsgenerator

11

Übersicht

Einleitung

Diese Kapitel beschreibt den Baustein FGEN.

Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Kurzbeschreibung	124
Darstellung	125
Parametrierung	126
Funktionsauswahl	127
Funktionsdefinition	128
Diagramme der einzelnen Funktionen	131
Sonderfälle	135
Taktdiagramme	136

Kurzbeschreibung

Funktionsbeschreibung

Der Funktionsbaustein FGEN stellt einen Funktionsgenerator dar. Er generiert am Ausgang Y eine Signalform, die in der Datenstruktur Para_FGEN definiert wird. Der Funktionsbaustein ist kaskadierbar, d. h. bei Anwendung mehrerer dieser EFBs können verschiedene Signalformen erzeugt und überlagert werden.

Folgende 8 verschiedene Signalformen können generiert werden:

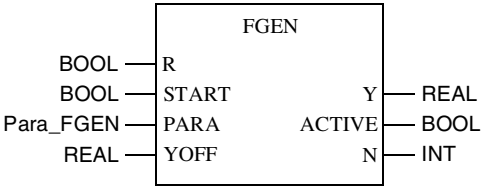
- Sprungfunktion
- Rampenfunktion
- Dreieckfunktion
- Sägezahnfunktion
- Rechteckfunktion
- Trapezfunktion
- Sinusfunktion
- Zufallszahl

Als zusätzliche Parameter können EN und ENO projektiert werden.

Darstellung

Symbol

Darstellung des Bausteins



Parameterbeschreibung FGEN

Beschreibung der Bausteinparameter

Parameter	Datentyp	Bedeutung
R	BOOL	"1": Rücksetzen
START	BOOL	1": Funktionsgenerator starten
PARA	Para_FGEN	Parameter
YOFF	REAL	Offset für Ausgang Y
Y	REAL	Ausgang des Funktionsgenerators
ACTIVE	BOOL	ACTIVE = 1: Funktionsgenerator ist aktiv
N	INT	Anzahl der Perioden seit Start

Parameterbeschreibung Para_FGEN

Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
func_no	INT	Wahl der Funktion (1 - 8) des Generators
amplitude	REAL	Amplitude der Funktion
halfperiod	TIME	Halbe Periodendauer
t_off	TIME	Ausschalt-Zeitkonstante
t_rise	TIME	Hochlauf-Zeitkonstante
t_acc	TIME	Verschleifzeit
unipolar	BOOL	"1 "= Signal unipolar "0" = Signal bipolar

Parametrierung

Reset

Der Parameter R bedeutet RESET. Ist dieser Parameter gesetzt (R = 1), so wird jede laufende Funktion damit sofort abgebrochen und am Ausgang Y wird der Wert des Parameters YOFF (Offset) ausgegeben. Gleichzeitig wird der Zähler für die Anzahl der Perioden N auf 0 zurückgestellt und ACTIVE auf "0" gesetzt.

Start des Funktions-generators

Mit dem Parameter START wird die über die Datenstruktur definierte Funktion gestartet (START = 1). Mit dem Beginn einer neuen Periode wird der Ausgang N inkrementiert. Wird der Parameter START auf 0 zurückgenommen, so wird die angefangene Periode der ausgewählten Funktion bis zum Ende bearbeitet. Solange wie eine Funktion läuft, ist der Ausgang ACTIVE auf 1. Ist die Periode beendet, so wird auch der Ausgang ACTIVE auf 0 zurückgesetzt.

Offset

Die über den Funktionsgenerator erzeugten Signalverläufe haben eine Amplitude mit dem Betrag von "amplitude", d.h. der Wertebereich reicht von -"amplitude" bis "amplitude" bei bipolarem Betrieb (unipolar = 0) bzw. von 0 bis "amplitude" bei unipolarem Betrieb (unipolar = 1). Über den Parameter YOFF kann diese Funktion gegenüber dem Nullpunkt verschoben werden.

Hinweis: Wird am Parameter YOFF der Ausgang eines anderen Funktionsgenerators angelegt, so überlagern sich die durch die beiden Funktionsgeneratoren erzeugten Signalverläufe.

Hochlaufzeit t_rise

Die Hochlaufzeit t_rise wird nur bei den Funktionen "Rampe" und "Trapez" verwendet. Bei der Funktion "Sägezahn" ergibt sich die Hochlaufzeit aus halfperiod - t_off. Bei der Funktion "Dreieck" ergibt sich die Hochlaufzeit aus 0.5 * (halfperiod - t_off).

Funktionsauswahl

Auswahl

Es gibt insgesamt 8 Funktionen, die mit dem Funktionsgenerator erzeugt werden können. Die Funktionsauswahl wird über func_no ausgewählt. Bei einem Funktionswechsel wird immer die zuletzt angewählte Funktion vollständig bearbeitet.

Folgende Funktionsnummern sind zulässig:

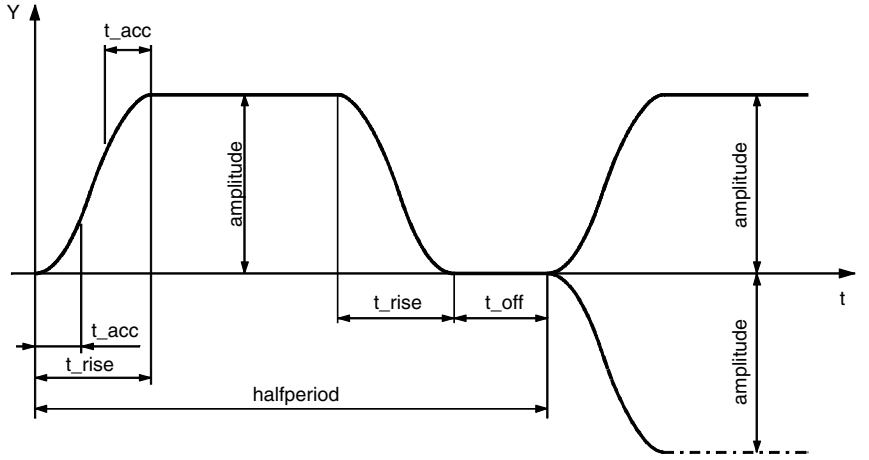
func_no	**Funktion**
1	Sprung
2	Rampe
3	Sägezahn
4	Dreieck
5	Rechteck
6	Trapez
7	Sinus
8	Zufallszahl

Funktionsdefinition

Definition

Die Funktion wird komplett in der Datenstruktur Para_FGEN definiert. Zunächst muß die Signalform festgelegt werden (siehe *Funktionsauswahl, S. 127*).

Als Grundtyp für die Definition wird das Trapez (Dreieck, Sägezahn, Rechteck)unipolar/bipolar gewählt).



Im Parameter amplitude wird die Amplitude der Funktion festgelegt. Hierbei ist zu beachten, daß diese Angabe für den unipolaren Betrieb gültig ist. Bei bipolarem Betrieb setzt sich die Amplitude aus amplitude und -amplitude zusammen, ist also doppelt so hoch.

Der Parameter halfperiod definiert die halbe Periodendauer.

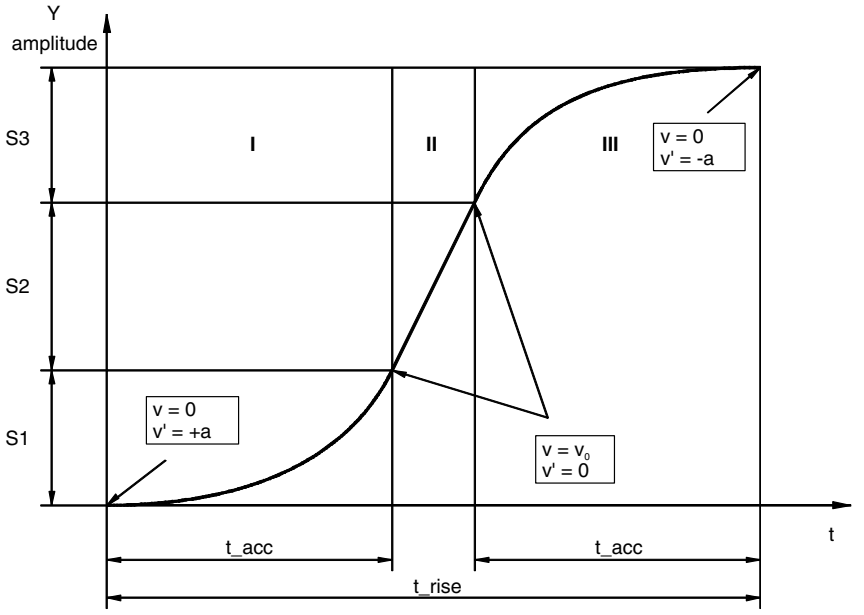
Der Parameter t_off definiert eine Ausschaltzeit (Ruhezeit). Eine Halbperiode der Funktion wird dann innerhalb der Zeit halfperiod - t_off ausgegeben.

Für die Definition der Trapezfunktion gibt es noch die Hochlaufzeit t_rise. Das ist die Zeit, in der das Signal von 0 auf amplitude hochlaufen soll. Diese Zeit wird auch für den Abfall von amplitude auf 0 benutzt.

"Verschleifen" einer Funktion

Soll eine Funktion in Form einer Rampe ansteigen oder abfallen, so sind die Übergänge zunächst immer mit einem Knick behaftet. Die Steigung ist hierbei nicht stetig. Zur Realisierung eines sanften Anstieg bzw. Abfall wird der "Verschliff" verwendet, d.h. die Rampe wird zu einer S-Kurve.

Verschleifen" einer Funktion



Diese wird in drei Teile eingeteilt. Im Teil I wird von 0 aus "beschleunigt" hochgefahren. Im Teil II wird mit der am Ende von Teil I erreichten Geschwindigkeit weitergefahren. Im Teil III wird mit der Beschleunigung aus Teil I gebremst, um sanft den Endpunkt anzufahren. Die Größe der Abschnitte sind frei wählbar. Sie werden durch Angabe von t_acc und t_rise definiert.

Die verwendete Beschleunigung wird nach folgenden Formeln berechnet:

$$\text{amplitude} = S1 + S2 + S3$$

mit

$$S3 = S1 = \frac{a}{2} \times t_acc^2$$

und

$$S2 = a \times t_acc \times (t_rise - 2 \times t_acc)$$

Daraus ergibt sich

$$a = \frac{\text{amplitude}}{t_acc \times t_rise - t_acc^2}$$

Hinweis: Verschleifen wird nur bei den Funktionen "Rampe", "Sägezahn", "Dreieck" und "Trapez" verwendet. "Sprung", "Rechteck" und "Sinus" sind keine "verschleifbaren" Funktionen.

Verwendung der einzelnen Parameter

Nutzung der Parameter innerhalb der verschiedenen Funktionen

Funktion	amplitude	halfperiod	t_off	t_rise	t_acc	uni-polar
Sprung	X					
Rampe	X			X	X	
Sägezahn	X	X	X	halfperiod - t_acc	X	X
Dreieck	X	X	X	(halfperiod - t_acc)/2	X	X
Rechteck	X	X	X			X
Trapez	X	X	X	X	X	X
Sinus	X	X	X			X
Zufallszahl	X					X

Die grafische Darstellung der Funktionen finden Sie im Abschnitt *Diagramme der einzelnen Funktionen, S. 131*

Unipolarer Betrieb

Über den Parameter unipolar kann definiert werden, ob die Funktion als unipolare oder als bipolare Funktion ausgegeben werden soll. Hierbei ist zu beachten, daß bei unipolarem Betrieb eine Periode trotzdem durch 2 "unipolare" Halbwellen gekennzeichnet ist.

Änderung von Funktions-parametern

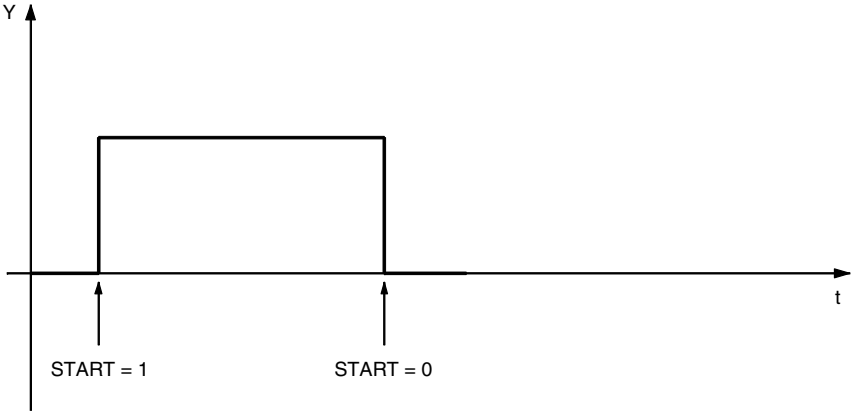
Während einer laufenden Periode können sämtliche Parameter der Funktion geändert werden. Die Änderungen werden jedoch erst nach abgeschlossener Periode wirksam. Wird beispielsweise die Ausschaltzeit t_off bei laufender Periode geändert, so wirkt sich dies erst in der darauf folgenden Periode aus.

Änderung der Funktion

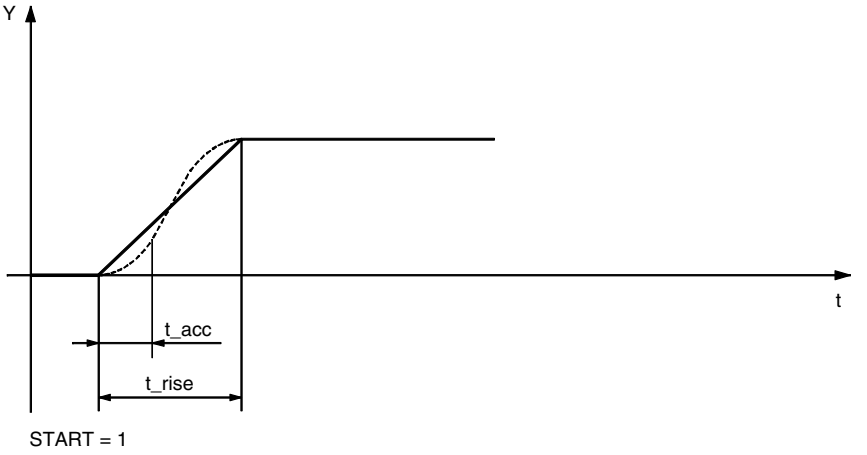
Wird der Parameter func_no während einer laufenden Periode geändert, so wird ebenfalls zunächst die aktuelle Periode mit der vorher gewählten Funktion(snummer) beendet. Danach wird die neue Funktion gestartet. Hierbei wird der Ausgang N, der die Anzahl der Perioden angibt zurückgesetzt.

Diagramme der einzelnen Funktionen

Sprungfunktion Darstellung der Sprungfunktion

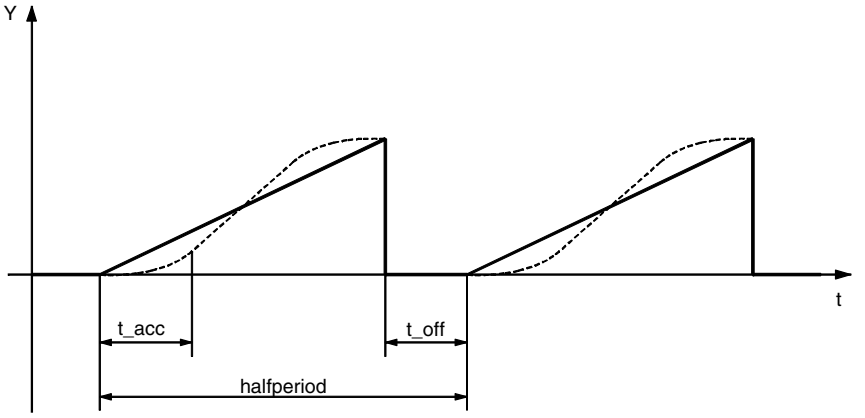


Rampenfunktion Darstellung der Rampenfunktion



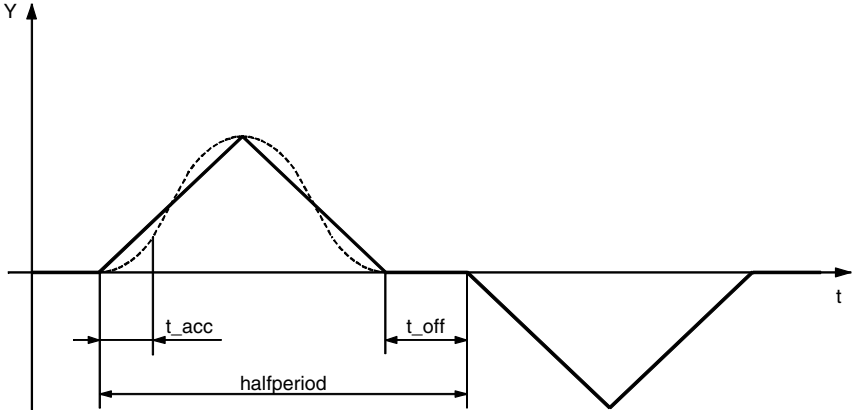
Sägezahn-funktion

Darstellung der Sägezahnfunktion



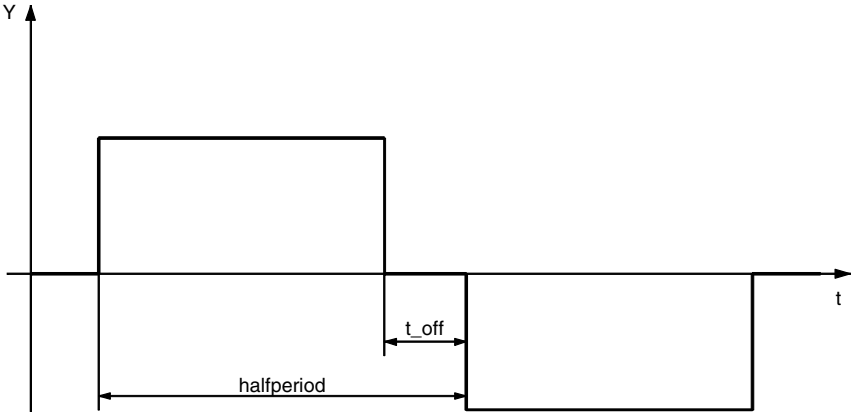
Dreiecksfunktion

Darstellung der Dreiecksfunktion



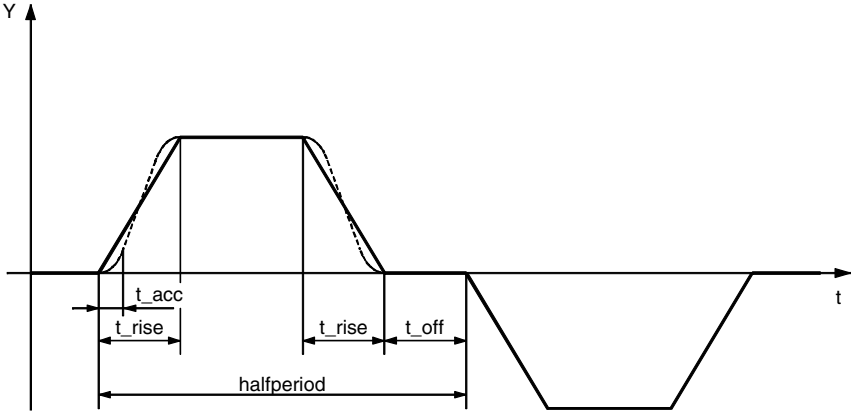
Rechteck-funktion

Darstellung der Rechteckfunktion

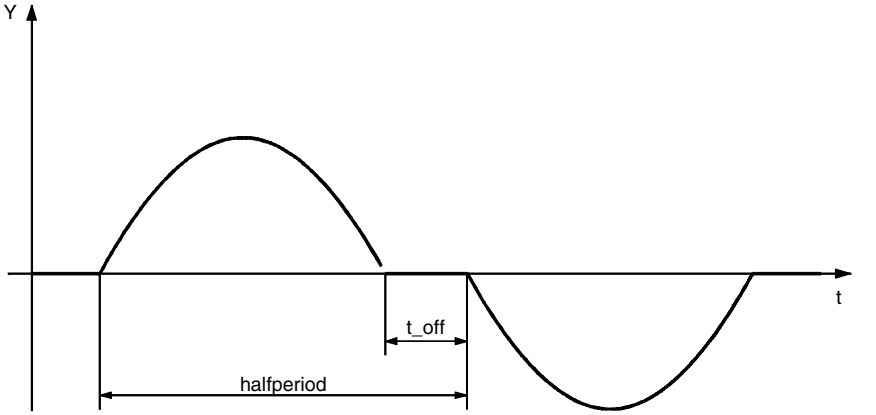


Trapezfunktion

Darstellung der Trapezfunktion



Sinusfunktion Darstellung der Sinusfunktion



Sonderfälle

Sprung-Funktion

Bei der Funktion "Sprung" wird der Ausgang

bei START = 0 auf den Wert Y = YOFF

und

bei START = 1 auf den Wert Y = YOFF + amplitude

gesetzt

Die Zeitangaben (t_off, t_rise, t_acc) spielen bei dieser Funktion keine Rolle.

Bei jeder neuen 0 → 1-Flanke am Eingang START wird der Ausgang N inkrementiert.

Es gibt für diese Funktion keinen bipolaren Betrieb, d.h. der Wert im Parameter unipolar wird nicht berücksichtigt.

Rampen-Funktion

Bei der Funktion "Rampe" wird der Ausgang Y von YOFF auf YOFF + amplitude rampenförmig hochgefahren. Solange der START dann weiterhin auf 1 liegt, bleibt der Ausgang Y auf dem Wert YOFF + amplitude stehen. Wird START auf 0 zurückgenommen, so springt der Ausgang Y auf den Wert YOFF.

Das Hochfahren wird durch die Zeit t_rise und t_acc bestimmt. Die Zeit t_rise gibt hierbei die Zeit an, in der der Wert von Y = YOFF auf den Wert Y = YOFF + amplitude hochfährt. Über t_acc kann diese Zeit "verschliffen" werden.

Bei jeder neuen 0 → 1-Flanke am Eingang START wird der Ausgang N inkrementiert.

Es gibt für diese Funktion keinen bipolaren Betrieb, d.h. der Wert im Parameter unipolar wird nicht berücksichtigt.

Zufallszahl

Bei der Funktion "Zufallszahl" wird der Ausgang Y mit einer sich "zufällig" ergebenden Zahl zwischen

YOFF ≤ Y ≤ YOFF + amplitude, bei unipolarem Betrieb

und

YOFF - amplitude ≤ Y ≤ YOFF + amplitude, bei bipolarem Betrieb

belegt.

Die Zeitangaben (t_off, t_rise, t_acc) spielen bei dieser Funktion keine Rolle.

Bei jeder neuen 0 → 1-Flanke am Eingang START wird der Ausgang N inkrementiert.

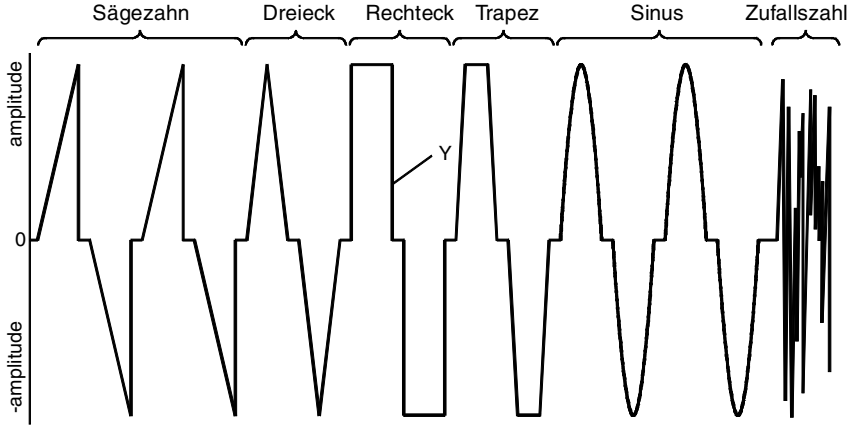
Taktdiagramme

Bipolarer Betrieb

Für die Darstellung der verschiedenen Funktionen im bipolaren Betrieb gelten folgende Parametervorgaben:

Parameter	Vorgabe
amplitude	1
halfperiod	10
t_off	2
t_rise	2
t_acc	0
unipolar	0

Bipolarer Betrieb

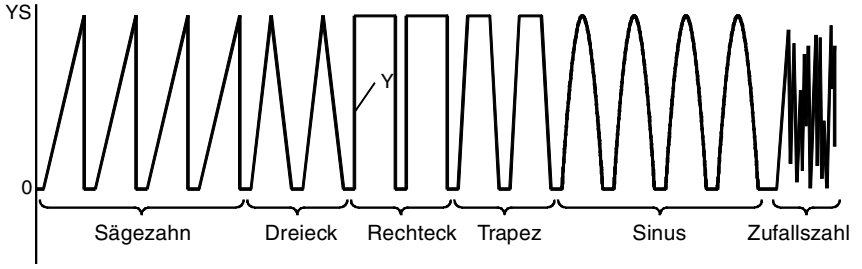


Unipolarer Betrieb

Für die Darstellung der verschiedenen Funktionen im unipolaren Betrieb gelten folgende Parametervorgaben:

Parameter	Vorgabe
amplitude	1
halfperiod	10
t_off	2
t_rise	2
t_acc	0
unipolar	1

Unipolarer Betrieb

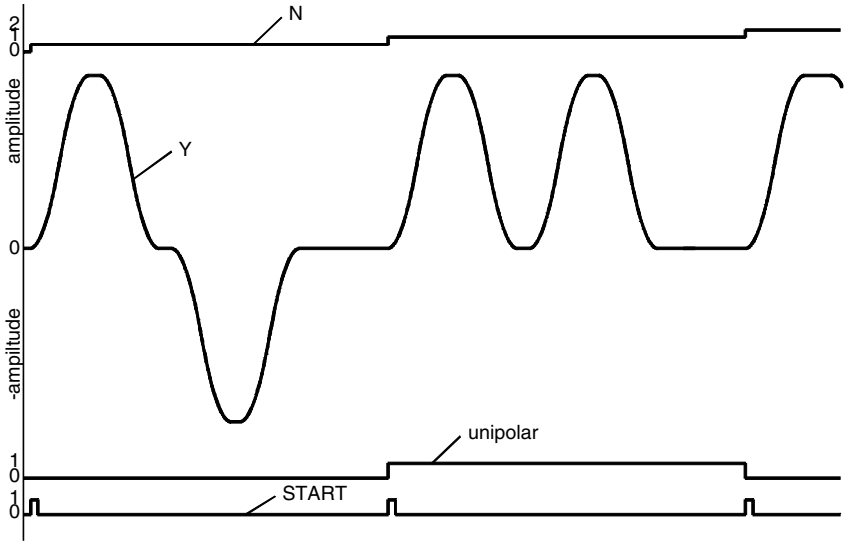


Trapez-Funktion

Für die Darstellung der Trapez-Funktion mit Verschleifzeit gelten folgende Parametervorgaben:

Parameter	Vorgabe
amplitude	1
halfperiod	10
t_off	1
t_rise	4
t_acc	1.5

Trapez-Funktion



INTEG: Integrierer mit Begrenzung

12

Übersicht

Einleitung

Diese Kapitel beschreibt den Baustein INTEG.

Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Kurzbeschreibung	140
Darstellung	141
Detailbeschreibung	142
Laufzeitfehler	143

Kurzbeschreibung

Funktionsbeschreibung

Der Funktionsbaustein realisiert einen Integrierer mit Begrenzung.

Der Funktionsbaustein besitzt folgende Eigenschaften:

- Betriebsarten Hand, Halt, Automatik
- Stellgrößen-Begrenzung im Automatik-Betrieb

Als zusätzliche Parameter können EN und ENO projektiert werden.

Formel

Die Übertragungsfunktion lautet:

$$G(s) = \frac{gain}{s}$$

Die Berechnungsformel lautet:

$$Y = Y_{(old)} + gain \times dt \times \frac{X_{(new)} + X_{(old)}}{2}$$

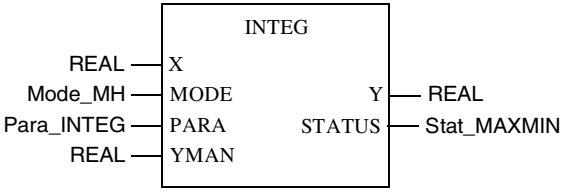
Bedeutung der Größen

Größe	Bedeutung
$X_{(old)}$	Wert des Eingangs X aus dem vorherigen Zyklus
$Y_{(old)}$	Wert des Ausgangs Y aus dem vorherigen Zyklus
dt	Zeitdifferenz zwischen aktuellem Zyklus und vorherigem Zyklus

Darstellung

Symbol

Darstellung des Bausteins



Parameterbeschreibung INTEG

Beschreibung der Bausteinparameter

Parameter	Datentyp	Bedeutung
X	REAL	Eingangsgröße
MODE	Mode_MH	Betriebsarten
PARA	Para_INTEG	Parameter
YMAN	REAL	Hand-Stellwert
Y	REAL	Ausgang
STATUS	Stat_MAXMIN	Status des Ausgangs

Parameterbeschreibung Mode_MH

Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
man	BOOL	"1" = Betriebsart Hand
halt	BOOL	"1" = Betriebsart Halt

Parameterbeschreibung Para_INTEG

Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
gain	REAL	Integrierverstärkung (Einheiten/Sekunde)
ymax	REAL	Obere Stellgrenze
ymin	REAL	Untere Stellgrenze

Parameterbeschreibung Stat_MAXMIN

Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
qmin	BOOL	"1" = Y hat untere Stellgrenze erreicht
qmax	BOOL	"1" = Y hat obere Stellgrenze erreicht

Detailbeschreibung

Parametrierung

Die Parametrierung des Funktionsbausteins erfolgt durch Festlegen der Integrierverstärkung gain und der Grenzwerte ymax und ymin für den Ausgang Y.

Die Werte ymax und ymin begrenzen den Ausgang nach oben sowie nach unten. Damit gilt ymin ≤ Y ≤ ymax

Das Erreichen der Grenzwerte, sowie eine Begrenzung des Ausgangssignals wird durch qmax und qmin angezeigt:

- qmax = 1 wenn Y ≥ ymax
- qmin = 1 wenn Y ≤ ymin

Betriebsarten

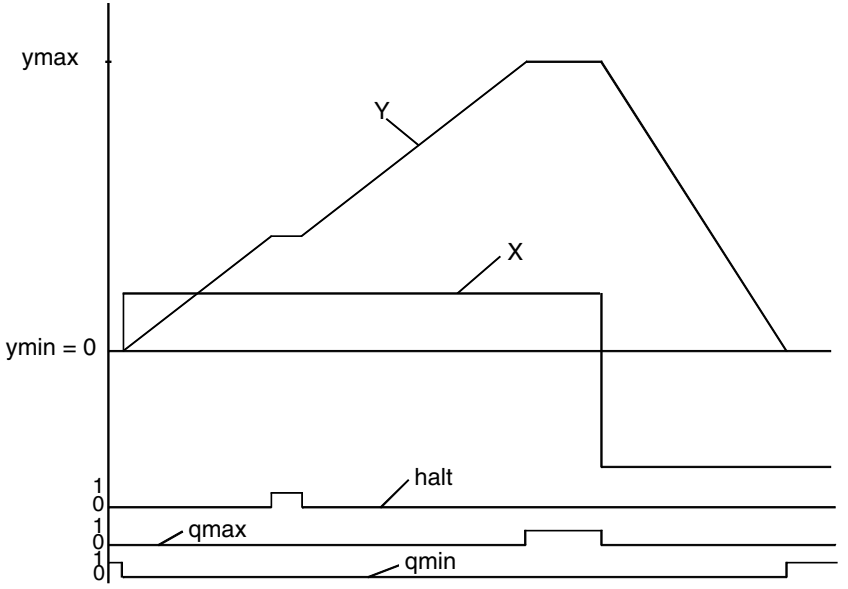
Es gibt drei Betriebsarten, die über die Elemente man und halt ausgewählt werden:

Betriebsart	man	halt	Bedeutung
Automatik	0	0	Der Funktionsbaustein wird, wie in "Parametrierung" beschrieben, bearbeitet.
Hand	1	0 oder 1	Der Handwert YMAN wird fest auf den Ausgang Y übertragen. Der Stellausgang wird jedoch durch ymax und ymin begrenzt.
Halt	0	1	Der Ausgang Y wird auf dem zuletzt berechneten Wert gehalten. Der Ausgang wird nicht mehr verändert, kann jedoch vom Anwender überschrieben werden.

Beispiel

Das Eingangssignal wird über die Zeit integriert. Im Fall eines Sprungs am Eingang X steigt (bei positiven Werten von X) bzw. fällt (bei negativen Werten von X) der Ausgang gemäß einer Rampenfunktion. Y liegt immer zwischen ymax und ymin; ist Y gleich ymax oder ymin wird dies entsprechend in qmax oder qmin angezeigt.

Darstellung der Sprungantwort des Integrierers



Laufzeitfehler

Fehlermeldung

Es erfolgt eine Fehlermeldung, wenn

- eine unzulässige Gleitkommazahl am Eingang YMAN oder X anliegt,
- ymax < ymin ist

INTEGRATOR: Integrierer mit Begrenzung

13

Übersicht

Einleitung

Diese Kapitel beschreibt den Baustein INTEGRATOR.

Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Kurzbeschreibung	146
Darstellung	147
Detailbeschreibung	148
Laufzeitfehler	149

Kurzbeschreibung

Funktionsbeschreibung

Der Funktionsbaustein realisiert einen Integrierer mit Begrenzung.

Der Funktionsbaustein besitzt folgende Eigenschaften:

- Betriebsarten Tracking und Automatik
- Stellgrößen-Begrenzung im Automatik-Betrieb

Als zusätzliche Parameter können EN und ENO projektiert werden.

Formel

Die Übertragungsfunktion lautet:

$$G(s) = \frac{GAIN}{s}$$

Die Formel für den Ausgang OUT lautet:

$$OUT = OUT_{(old))} + GAIN \times dt \times \frac{IN_{(new)} + IN_{(old)}}{2}$$

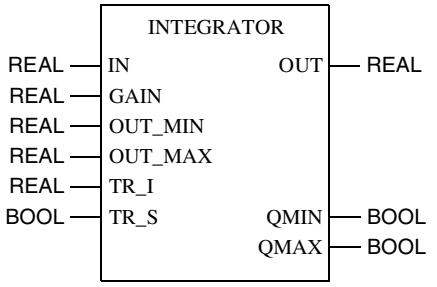
Bedeutung der Größen

Größe	Bedeutung
$IN_{(new)}$	aktueller Wert des Eingangs IN
$IN_{(old)}$	Wert des Eingangs IN aus dem vorherigen Zyklus
$OUT_{(old)}$	Wert des Ausgangs OUT aus dem vorherigen Zyklus
dt	Zeitdifferenz zwischen aktuellem Zyklus und vorherigem Zyklus

Darstellung

Symbol

Darstellung des Bausteins



Parameterbeschreibung

Beschreibung der Bausteinparameter

Parameter	Datentyp	Bedeutung
IN	REAL	Eingangsgröße
GAIN	REAL	Integrierverstärkung
OUT_MIN	REAL	Untere Stellgrenze
OUT_MAX	REAL	Obere Stellgrenze
TR_I	REAL	Initialisierungs-Eingang
TR_S	BOOL	Initialisierungs-Typ "1" = Betriebsart Tracking "0" = Betriebsart Automatik
OUT	REAL	Ausgang
QMIN	BOOL	"1" = Ausgang OUT hat untere Grenze erreicht
QMAX	BOOL	"1" = Ausgang OUT hat obere Grenze erreicht

Detailbeschreibung

Parametrierung

Die Parametrierung des Funktionsbausteins erfolgt durch Festlegen der Integrierverstärkung GAIN und der Grenzwerte OUT_MAX und OUT_MIN für den Ausgang OUT.

Die Grenzen OUT_MAX und OUT_MIN begrenzen den Ausgang nach oben sowie nach unten. Damit gilt OUT_MIN ≤ OUT ≤ OUT_MAX.

Das Erreichen der Grenzwerte, bzw. eine Begrenzung des Ausgangssignals wird durch die beiden Merker QMAX und QMIN angezeigt:

- QMAX = 1 wenn OUT ≥ OUT_MAX
- QMIN = 1 wenn OUT ≤ OUT_MIN

Betriebsarten

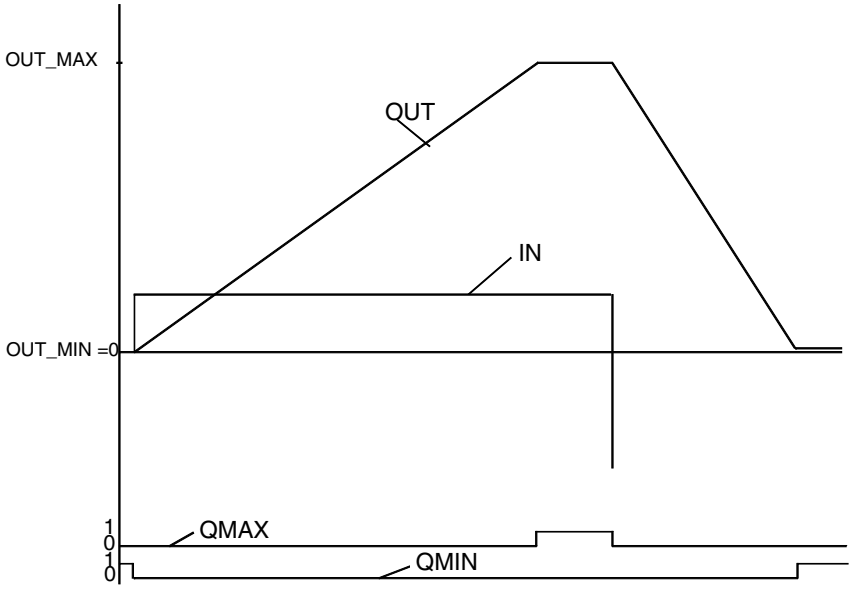
Es gibt zwei Betriebsarten, die über den Eingang TR_S ausgewählt werden

Betriebsart	TR_S	Bedeutung
Automatik	0	Der Funktionsbaustein wird, wie in "Parametrierung" beschrieben, bearbeitet.
Tracking	1	Der Trackingwert TR_I wird fest auf den Ausgang OUT übertragen. Der Stellausgang wird jedoch durch OUT_MAX und OUT_MIN begrenzt.

Beispiel

Das Eingangssignal wird über die Zeit integriert. Im Fall eines Sprungs am Eingang IN steigt (bei positiven Werten von IN) bzw. fällt (bei negativen Werten von IN) der Ausgang gemäß einer Rampenfunktion. OUT liegt immer zwischen OUTMAX und OUT_MIN; ist OUT gleich OUT_MAX oder OUT_MIN wird dies entsprechend in QMAX oder QMIN angezeigt.

Es ist die Sprungantwort des Integrierers dargestellt:.



Laufzeitfehler

Fehlermeldung

Bei OUT_MAX < OUT_MIN erfolgt eine Fehlermeldung.

INTEGRATOR1: Integrierer mit Begrenzung

14

Übersicht

Einleitung

Diese Kapitel beschreibt den Baustein INTEGRATOR1.

Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Kurzbeschreibung	152
Darstellung	153
Detailbeschreibung	154
Laufzeitfehler	155

Kurzbeschreibung

Funktionsbeschreibung

Der Funktionsbaustein realisiert einen Integrierer mit Begrenzung.

Der Funktionsbaustein besitzt folgende Eigenschaften:

- Betriebsarten Hand, Halt und Automatik
- Stellgrößen-Begrenzung im Automatik-Betrieb

Als zusätzliche Parameter können EN und ENO projektiert werden.

Formel

Die Übertragungsfunktion lautet:

$$G(s) = \frac{GAIN}{s}$$

Die Formel für den Ausgang Y lautet:

$$Y = Y_{(old))} + GAIN \times dt \times \frac{X_{(new)} + X_{(old)}}{2}$$

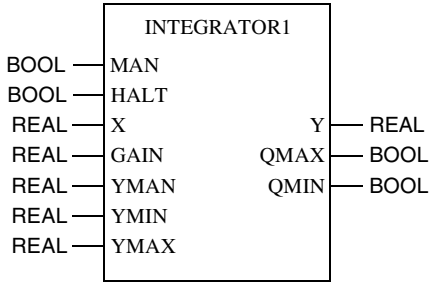
Bedeutung der Größen

Größe	Bedeutung
$X_{(old)}$	Wert des Eingangs X aus dem vorherigen Zyklus
$Y_{(old)}$	Wert des Ausgangs Y aus dem vorherigen Zyklus
dt	Zeitdifferenz zwischen aktuellem Zyklus und vorherigem Zyklus

Darstellung

Symbol

Darstellung des Bausteins



Parameterbeschreibung

Beschreibung der Bausteinparameter

Parameter	Datentyp	Bedeutung
MAN	BOOL	"1" = Betriebsart Hand
HALT	BOOL	"1" = Betriebsart Halt
X	REAL	Eingangsgröße
GAIN	REAL	Integrierverstärkung
YMAX	REAL	Obere Stellgrenze
YMIN	REAL	Untere Stellgrenze
YMAN	REAL	Hand-Stellwert
Y	REAL	Ausgang
QMAX	BOOL	"1" = Ausgang Y hat obere Grenze erreicht
QMIN	BOOL	"1" = Ausgang Y hat untere Grenze erreicht

Detailbeschreibung

Parametrierung

Die Parametrierung des Funktionsbausteins erfolgt durch Festlegen der Integrierverstärkung GAIN und der Grenzwerte YMAX und YMIN für den Ausgang Y.

Die Grenzen YMAX und YMIN begrenzen den Ausgang nach oben sowie nach unten. Damit gilt YMIN ≤ Y ≤ YMAX.

Das Erreichen der Grenzwerte, bzw. eine Begrenzung des Ausgangssignals wird durch die beiden Merker QMAX und QMIN angezeigt:

- QMAX = 1 wenn Y ≥ YMAX
- QMIN = 1 wenn Y ≤ YMIN

Betriebsarten

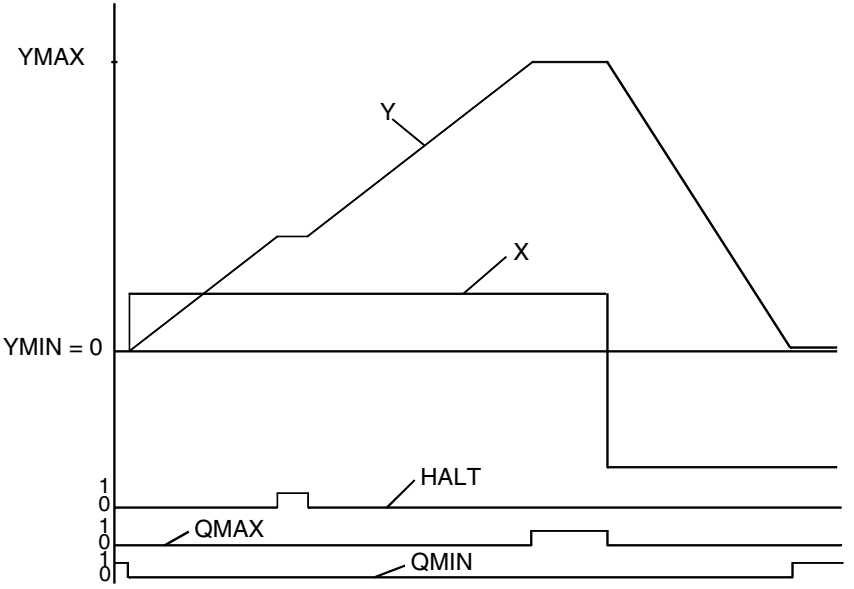
Es gibt drei Betriebsarten, die über die Eingänge MAN und HALT ausgewählt werden:

Betriebsart	MAN	HALT	Bedeutung
Automatik	0	0	Der Funktionsbaustein wird, wie in "Parametrierung" beschrieben, bearbeitet.
Hand	1	0 oder 1	Der Handwert YMAN wird fest auf den Ausgang Y übertragen. Der Stellausgang wird jedoch durch YMAX und YMIN begrenzt.
Halt	0	1	Der Ausgang Y wird auf dem zuletzt berechneten Wert gehalten.

Beispiel

Das Eingangssignal wird über die Zeit integriert. Im Fall eines Sprungs am Eingang X steigt (bei positiven Werten von X) bzw. fällt (bei negativen Werten von X) der Ausgang gemäß einer Rampenfunktion. Y liegt immer zwischen YMAX und YMIN; ist Y gleich YMAX oder YMIN wird dies entsprechend in QMAX oder QMIN angezeigt.

Es ist die Sprungantwort des Integrierers dargestellt:.



Laufzeitfehler

Fehlermeldung

Bei YMAN < YMIN erfolgt eine Fehlermeldung.

K_SQRT: Quadratwurzel

15

Übersicht

Einleitung

Diese Kapitel beschreibt den Baustein K_SQRT.

Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Kurzbeschreibung	158
Darstellung	158
Laufzeitfehler	159

Kurzbeschreibung

Funktionsbeschreibung

Dieser Funktionsbaustein berechnet die gewichtete Quadratwurzel eines numerischen Wertes. Es kann ein Trennpunkt definiert werden, unterhalb dessen der Funktionsbaustein den Wert Null ausgibt.

Das Ziehen der Quadratwurzel dient typischerweise zur Linearisierung eines Durchfluß-Meßwertes durch ein Drosselgerät.

Als zusätzliche Parameter können EN und ENO projektiert werden.

Formel

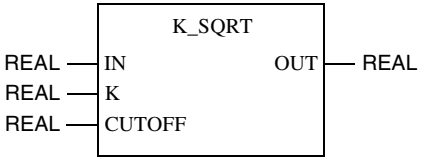
Der Funktionsbaustein führt folgende Berechnung durch:

Berechnung	Bedingung
$OUT = K\sqrt{IN}$	$IN \geq CUTOFF$
$OUT = 0$	$IN < 0$ oder $IN < CUTOFF$

Darstellung

Symbol

Darstellung des Bausteins



Parameterbeschreibung

Beschreibung der Bausteinparameter

Parameter	Datentyp	Bedeutung
IN	REAL	Zu verarbeitender numerischer Wert
K	REAL	Gewichtungskoeffizient
CUTOFF	REAL	Trennpunkt
OUT	REAL	Ergebnis der Berechnung

Laufzeitfehler

Fehlermeldung	Es wird ein Fehler angezeigt, wenn an einem der Eingänge ein nicht gleitender Wert erfaßt wird oder bei einer Berechnung in Gleitkommawerten ein Problem auftritt. Der Ausgang OUT bleibt in diesem Fall unverändert.
Warnung	Es wird eine Warnung ausgegeben, wenn der CUTOFF-Eingang negativ ist. Der Funktionsbaustein verwendet dann zur Berechnung den Wert 0.

LAG: Verzögerungsglied 1. Ordnung

16

Übersicht

Einleitung

Dieses Kapitel beschreibt den Baustein LAG.

Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Kurzbeschreibung	162
Darstellung	163
Detailbeschreibung	164

Kurzbeschreibung

Funktionsbeschreibung

Der Funktionsbaustein stellt ein Verzögerungsglied 1. Ordnung (Tiefpaß) dar

Der Funktionsbaustein besitzt folgende Betriebsarten:

- Hand
- Halt
- Automatik

Als zusätzliche Parameter können EN und ENO projektiert werden.

Formel

Die Übertragungsfunktion lautet:

$$G(s) = gain \times \frac{gain}{1 + s \times lag}$$

Die Berechnungsformel lautet:

$$Y = Y_{(old)} + \frac{dt}{lag + dt} \times \left(gain \times \frac{X_{(old)} + X_{(new)}}{2} - Y_{(old)} \right)$$

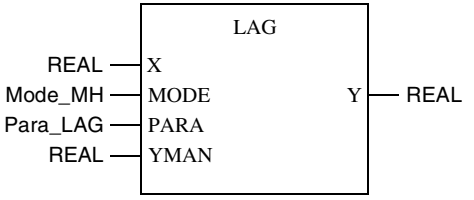
Bedeutung der Größen

Größe	Bedeutung
$X_{(old)}$	Wert des Eingangs X aus dem vorherigen Zyklus
$Y_{(old)}$	Wert des Ausgangs Y aus dem vorherigen Zyklus
dt	Zeitdifferenz zwischen aktuellem Zyklus und vorherigem Zyklus

Darstellung

Symbol

Darstellung des Bausteins



Parameterbeschreibung LAG

Beschreibung der Bausteinparameter

Parameter	Datentyp	Bedeutung
X	REAL	Eingangswert
MODE	Mode_MH	Betriebsarten
PARA	Para_LAG	Parameter
YMAN	REAL	Hand-Stellwert
Y	REAL	Ausgang

Parameterbeschreibung Mode_MH

Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
man	BOOL	"1" = Betriebsart Hand
halt	BOOL	"1" = Betriebsart Halt

Parameterbeschreibung Para_LAG

Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
gain	REAL	Verstärkungsfaktor
lag	TIME	Verzögerungszeitkonstante

Detailbeschreibung

Parametrierung

Die Parametrierung des Funktionsbausteins erfolgt durch Festlegen des Verstärkungsfaktors gain sowie der Parametrierung der Verzögerungszeitkonstante lag.

Dem Einheitssprung am Eingang X (Sprung am Eingang X von 0 auf 1.0) folgt der Ausgang Y verzögert nach. Er wird sich nach einer e-Funktion

$$\exp(-t/\mathrm{lag})$$

dem Wert $\mathrm{gain} \times \mathrm{X}$ nähern.

Betriebsarten

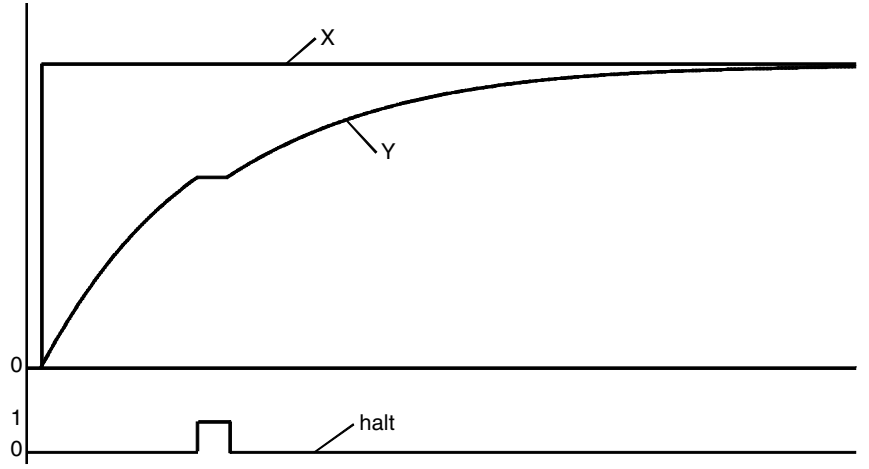
Es gibt drei Betriebsarten, die über die Elemente man und halt ausgewählt werden:

Betriebsart	man	halt	Bedeutung
Automatik	0	0	Der Funktionsbaustein wird, wie in "Parametrierung" beschrieben, bearbeitet.
Hand	1	0 oder 1	Der Handwert YMAN wird fest auf den Ausgang Y übertragen.
Halt	0	1	Der Ausgang Y wird auf dem zuletzt berechneten Wert gehalten. Der Ausgang wird nicht mehr verändert, kann jedoch vom Anwender überschrieben werden.

Beispiel

Das Diagramm zeigt ein Beispiel für die Sprungantwort des Funktionsbausteins: Der Eingang X springt auf einen neuen Wert und der Ausgang Y folgt in einer e-Funktion dem Eingang X.

Sprungantwort des Funktionsbausteins LAG bei gain = 1



LAG1: Verzögerungsglied 1. Ordnung

17

Übersicht

Einleitung

Dieses Kapitel beschreibt den Baustein LAG1.

Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Kurzbeschreibung	168
Darstellung	169
Detailbeschreibung	170

Kurzbeschreibung

Funktionsbeschreibung

Der Funktionsbaustein stellt ein Verzögerungsglied 1. Ordnung dar.

Der Funktionsbaustein besitzt folgende Betriebsarten:

- Hand
- Halt
- Automatik

Als zusätzliche Parameter können EN und ENO projektiert werden.

Formel

Die Übertragungsfunktion lautet:

$$G(s) = gain \times \frac{1}{1 + s \times lag}$$

Die Berechnungsformel lautet:

$$Y = Y_{(old)} + \frac{dt}{LAG + dt} \times \left(gain \times \frac{X_{(old)} + X_{(new)}}{2} - Y_{(old)} \right)$$

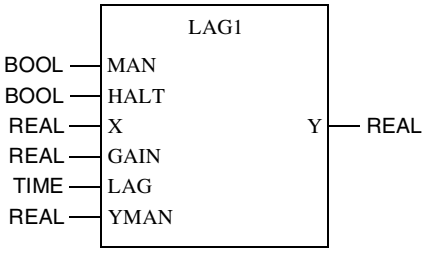
Bedeutung der Größen

Größe	Bedeutung
$X_{(old)}$	Wert des Eingangs X aus dem vorherigen Zyklus
$Y_{(old)}$	Wert des Ausgangs Y aus dem vorherigen Zyklus
dt	Zeitdifferenz zwischen aktuellem Zyklus und vorherigem Zyklus

Darstellung

Symbol

Darstellung des Bausteins



Parameterbeschreibung

Beschreibung der Bausteinparameter

Parameter	Datentyp	Bedeutung
MAN	BOOL	"1" = Betriebsart Hand
HALT	BOOL	"1" = Betriebsart Halt
X	REAL	Eingangswert
GAIN	REAL	Verstärkungsfaktor
LAG	TIME	Verzögerungszeitkonstante
YMAN	REAL	Hand-Stellwert
Y	REAL	Ausgang

Detailbeschreibung

Parametrierung

Die Parametrierung des Funktionsbausteins erfolgt durch Festlegen des Verstärkungsfaktors GAIN sowie der Parametrierung der Verzögerungszeitkonstante LAG.

Dem Einheitssprung am Eingang X (Sprung am Eingang X von 0 auf 1.0) folgt der Ausgang Y verzögert nach. Er wird sich nach einer e-Funktion

$$\exp(-t/(\mathrm{LAG}))$$

dem Wert $\mathrm{GAIN} \times \mathrm{X}$ nähern.

Betriebsarten

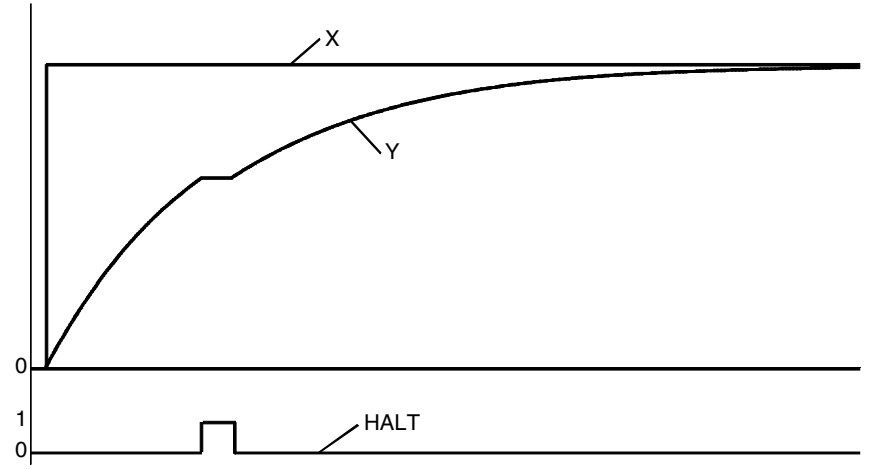
Es gibt drei Betriebsarten, die über die Eingänge MAN und HALT ausgewählt werden:

Betriebsart	MAN	HALT	Bedeutung
Automatik	0	0	Der Funktionsbaustein wird, wie in "Parametrierung" beschrieben, bearbeitet.
Hand	1	0 oder 1	Der Handwert YMAN wird fest auf den Ausgang Y übertragen.
Halt	0	1	Der Ausgang Y wird auf dem zuletzt berechneten Wert gehalten.

Beispiel

Das Diagramm zeigt ein Beispiel für die Sprungantwort des PLAG-Gliedes: der Eingang X springt auf einen neuen Wert und der Ausgang Y folgt in einer e-Funktion dem Eingang X.

Sprungantwort des Funktionsbausteins LAG1 bei GAIN = 1



LAG2: Verzögerungsglied 2. Ordnung

18

Übersicht

Einleitung

Dieses Kapitel beschreibt den Baustein LAG2.

Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Kurzbeschreibung	174
Darstellung	175
Detailbeschreibung	176
Taktdiagramme	177

Kurzbeschreibung

Funktionsbeschreibung

Der Funktionsbaustein LAG2 stellt ein Verzögerungsglied 2. Ordnung dar.

Der Funktionsbaustein besitzt folgende Betriebsarten:

- Hand
- Halt
- Automatik

Als zusätzliche Parameter können EN und ENO projektiert werden.

Formel

Die Übertragungsfunktion lautet:

$$G(s) = gain \times \frac{1}{1 + s \times 2 \times \frac{dmp}{freq} + \left(\frac{s}{freq}\right)^2}$$

Die Berechnungsformel lautet:

$$Y_{(new)} = A \times B$$

wobei

$$A = \frac{gain \times X \times (freq \times dt)^2 + Y_{(old)}}{1 + 2 \times dmp \times freq \times dt + (freq \times dt)^2}$$

und

$$B = \frac{(2 \times dmp \times freq \times dt \times 2) - Y_{(old2)}}{1 + 2 \times dmp \times freq \times dt + (freq \times dt)^2}$$

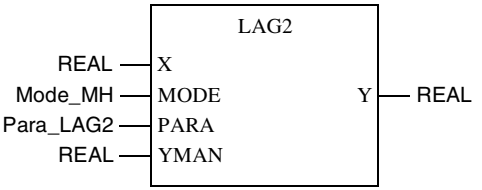
Bedeutung der Größen

Größe	Bedeutung
$Y_{(old)}$	Wert des Ausgangs Y aus dem vorherigen Zyklus
$Y_{(old2)}$	Wert des Ausgangs Y aus dem vorvorherigen Zyklus
dt	Zeitdifferenz zwischen aktuellem Aufruf und vorherigem Aufruf

Darstellung

Symbol

Darstellung des Bausteins



Parameterbeschreibung LAG2

Beschreibung der Bausteinparameter

Parameter	Datentyp	Bedeutung
X	REAL	Eingangswert
MODE	Mode_MH	Betriebsart
PARA	Para_LAG2	Parameter
YMAN	REAL	Handwert für den Ausgang
Y	REAL	Ausgang

Parameterbeschreibung Mode_MH

Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
man	BOOL	"1" = Betriebsart Hand
halt	BOOL	"1" = Betriebsart Halt

Parameterbeschreibung Para_LAG2

Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
gain	REAL	Verstärkungsfaktor
dmp	REAL	Dämpfung
freq	REAL	Eigenkreisfrequenz

Detailbeschreibung

Parametrierung

Die Parametrierung des Funktionsbausteins erfolgt durch Festlegen des Verstärkungsfaktors gain sowie der Parametrierung der Dämpfung dmp und der Eigenkreisfrequenz freq.

Die Dämpfung dmp und die Eigenkreisfrequenz freq sind positiv einzugeben.

Dem Sprung am Eingang X folgt der Ausgang Y in einer gedämpften Schwingung nach. Die Periodendauer der ungedämpften Schwingung ist T = 1/freq. Für Dämpfungswerte dmp < 1 spricht man von einer gedämpften Schwingung. Für Dämpfungswerte ≥ 1 spricht man von aperiodischem Verhalten (d.h. ohne Schwingung); in diesem Fall folgt der Ausgang dem Eingang ähnlich wie bei 2 LAG Funktionsbausteinen, die in Reihe geschaltet sind.

Betriebsarten

Es gibt drei Betriebsarten, die über die Elemente man und halt ausgewählt werden:

Betriebsart	man	halt	Bedeutung
Automatik	0	0	Der Funktionsbaustein wird, wie in "Parametrierung" beschrieben, bearbeitet.
Hand	1	0 oder 1	Der Handwert YMAN wird fest auf den Ausgang Y übertragen.
Halt	0	1	Der Ausgang Y wird auf dem zuletzt berechneten Wert gehalten. Der Ausgang wird nicht mehr verändert, kann jedoch vom Anwender überschrieben werden.

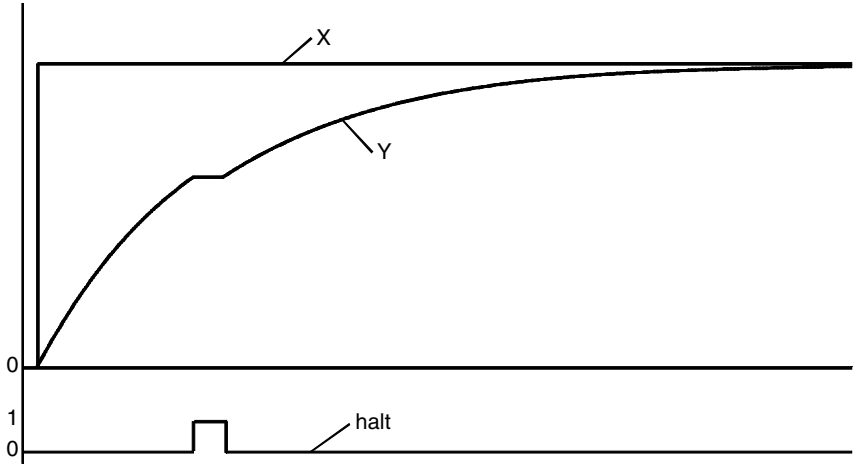
Taktdiagramme

Übersicht

Die folgenden Diagramme zeigen Beispiele für die Sprungantwort des LAG2-Gliedes bei unterschiedlicher Parametrierung.

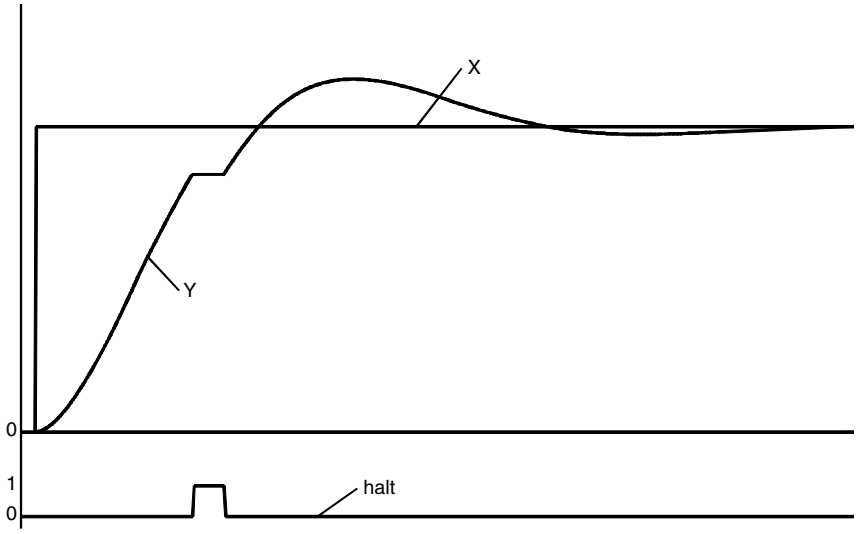
Dämpfung dmp = 1

Bei einer Dämpfung von dmp = 1 folgt der Ausgang Y dem Eingang X in einem aperiodenen Vorgang.



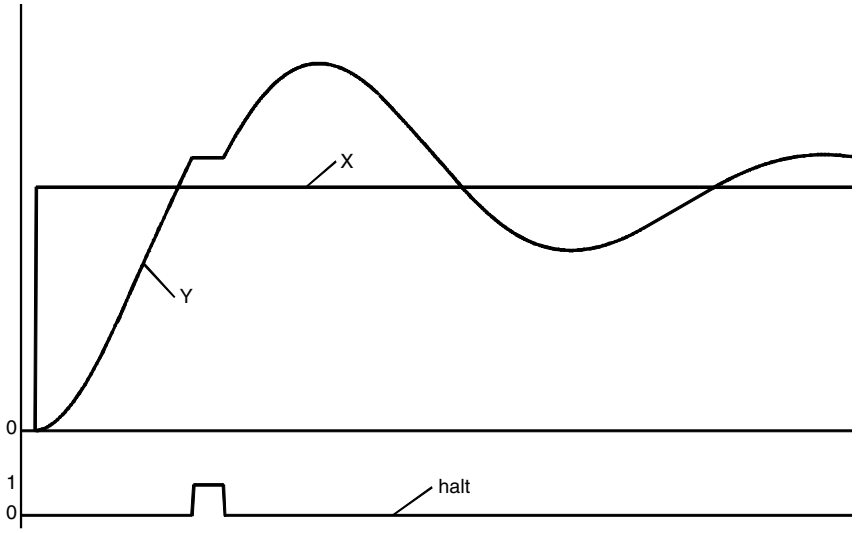
Dämpfung dmp = 0.5

Bei einer Dämpfung von dmp = 0.5 folgt der Ausgang Y dem Eingang X in einem gedämpft periodischen Vorgang.



Dämpfung dmp = 0.2

Bei einer Dämpfung von dmp = 0.2 erkennt man deutlich, daß die Sprungantwort wesentlich weniger gedämpft ist.



LAG_FILTER: Verzögerungsglied 1. Ordnung

19

Übersicht

Einleitung

Dieses Kapitel beschreibt den Baustein LAG_Filter.

Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Kurzbeschreibung	180
Darstellung	181
Detailbeschreibung	182

Kurzbeschreibung

Funktionsbeschreibung

Der Funktionsbaustein stellt ein Verzögerungsglied 1. Ordnung dar.

Der Funktionsbaustein besitzt folgende Betriebsarten:

- Tracking
- Automatik

Als zusätzliche Parameter können EN und ENO projektiert werden.

Formel

Die Übertragungsfunktion lautet:

$$G(s) = GAIN \times \frac{1}{1 + s \times LAG}$$

Die Berechnungsformel lautet:

$$OUT = OUT_{(old)} + \frac{dt}{LAG + dt} \times \left(GAIN \times \frac{IN_{(old)} + IN_{(new)}}{2} - OUT_{(old)} \right)$$

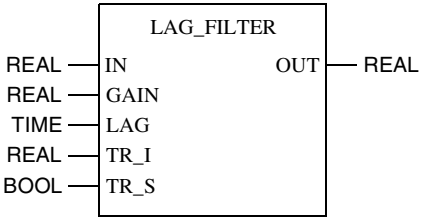
Bedeutung der Größen

Größe	Bedeutung
$IN_{(old)}$	Wert des Eingangs IN aus dem vorherigen Zyklus
$OUT_{(old)}$	Wert des Ausgangs OUT aus dem vorherigen Zyklus
dt	Zeitdifferenz zwischen aktuellem Zyklus und vorherigem Zyklus

Darstellung

Symbol

Darstellung des Bausteins



Parameterbeschreibung

Beschreibung der Bausteinparameter

Parameter	Datentyp	Bedeutung
IN	REAL	Eingangswert
GAIN	REAL	Verstärkungsfaktor
LAG	TIME	Verzögerungszeitkonstante
TR_I	REAL	Initialisierungs-Eingang
TR_S	BOOL	Initialisierungs-Typ "1" = Betriebsart Tracking "0" = Betriebsart Automatik
OUT	REAL	Ausgang

Detailbeschreibung

Parametrierung

Die Parametrierung des Funktionsbausteins erfolgt durch Festlegen des Verstärkungsfaktors GAIN sowie der Parametrierung der Verzögerungszeitkonstante LAG.

Dem Einheitssprung am Eingang IN (Sprung am Eingang IN von 0 auf 1.0) folgt der Ausgang OUT verzögert nach. Er wird sich nach einer e-Funktion

$$\exp(-t/\mathrm{LAG})$$

dem Wert $\mathrm{GAIN} \times \mathrm{X}$ nähern.

Betriebsarten

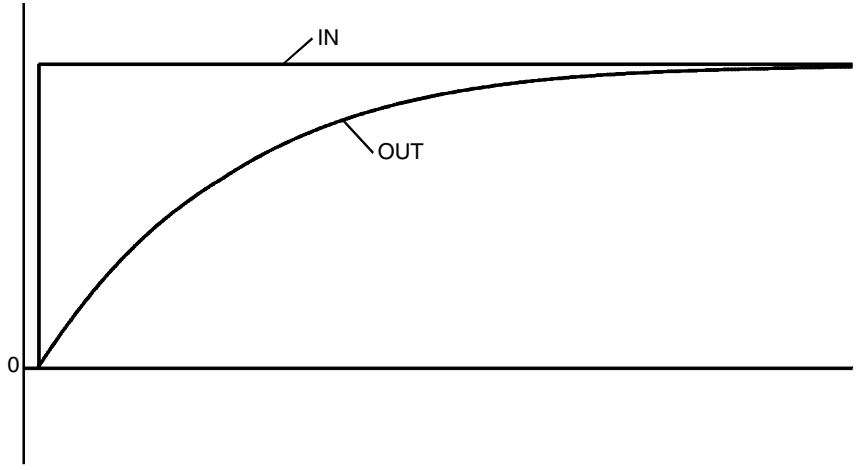
Es gibt zwei Betriebsarten, die über den Eingang TR_S ausgewählt werden:

Betriebsart	TR_S	Bedeutung
Automatik	0	Der Funktionsbaustein wird, wie in "Parametrierung" beschrieben, bearbeitet.
Tracking	1	Der Trackingwert TR_I wird fest auf den Ausgang OUT übertragen.

Beispiel

Das Diagramm zeigt ein Beispiel für die Sprungantwort des LAG_FILTER-Funktionbausteins: der Eingang IN spring auf einen neuen Wert und der Ausgang OUT folgt in einer e-Funktion dem Eingang IN.

Sprungantwort des Funktionsbausteins LAG_FILTER bei GAIN = 1



LDLG: PD-Glied mit Glättung

20

Übersicht

Einleitung

Dieses Kapitel beschreibt den Baustein LDLG.

Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Kurzbeschreibung	184
Darstellung	185
Detailbeschreibung	186
Beispiele zum Funktionsbaustein LDLG	187

Kurzbeschreibung

Funktionsbeschreibung

Der Funktionsbaustein stellt ein PD-Glied mit nachfolgender Glättung dar.

Der Funktionsbaustein besitzt folgende Eigenschaften:

- Definierbare Verzögerung des D-Anteils
- Betriebsarten Tracking, Automatik

Als zusätzliche Parameter können EN und ENO projektiert werden.

Formel

Die Übertragungsfunktion lautet:

$$G(s) = GAIN \times \frac{1 + s \times LEAD}{1 + s \times LAG}$$

Die Berechnungsformel lautet:

$$OUT = \frac{LAG \times OUT_{(old)} + GAIN \times ((LEAD + dt) \times IN - LEAD \times IN_{(old)})}{LAG + dt}$$

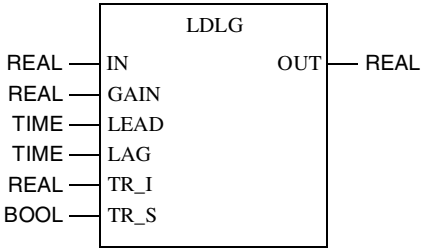
Bedeutung der Größen

Größe	Bedeutung
$IN_{(old)}$	Wert des Eingangs IN aus dem vorherigen Zyklus
$OUT_{(old)}$	Wert des Ausgangs OUT aus dem vorherigen Zyklus
dt	Zeitdifferenz zwischen aktuellem Zyklus und vorherigem Zyklus

Darstellung

Symbol

Darstellung des Bausteins



Parameterbeschreibung

Beschreibung der Bausteinparameter

Parameter	Datentyp	Bedeutung
IN	REAL	Eingang
GAIN	REAL	Verstärkungsfaktor
LEAD	TIME	Differenzierzeitkonstante
LAG	TIME	Verzögerungszeitkonstante
TR_I	REAL	Initialisierung-Eingang
TR_S	BOOL	Initialisierung-Typ "1" = Betriebsart Tracking "0" = Betriebsart Automatik
OUT	REAL	Ausgang

Detailbeschreibung

Parametrierung

Die Parametrierung des Funktionsbausteins erfolgt durch Festlegen des Verstärkungsfaktors GAIN sowie der Parametrierung der Differenzierzeitkonstante LEAD und der Verzögerungszeitkonstante LAG.

Für sehr kleine Abtastzeiten und dem Einheitssprung am Eingang IN (Sprung am Eingang IN von 0 auf 1.0) wird der Ausgang OUT auf den Wert $GAIN \times LEAD/LAG$ springen (theoretischer Wert - real ein bißchen kleiner, bedingt durch die nicht unendlich kleine Abtastzeit), um dann mit der Zeitkonstante LAG auf den Wert $GAIN \times 1.0$ zuzugehen.

Betriebsarten

Es gibt zwei Betriebsarten, die über den Eingang TR_S ausgewählt werden:

Betriebsart	TR_S	Bedeutung
Automatik	0	Der Funktionsbaustein wird, wie in "Parametrierung" beschrieben, bearbeitet.
Tracking	1	Der Trackingwert TR_I wird fest auf den Ausgang OUT übertragen.

Beispiele zum Funktionsbaustein LDLG

Beispiel-Übersicht

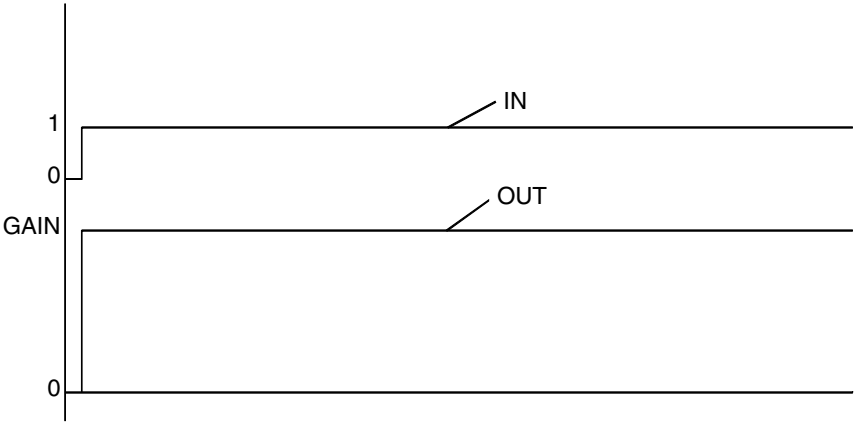
Folgende Beispiele werden in den nachfolgenden Diagrammen dargestellt:

- LEAD = LAG
- LEAD/LAG = 0.5, GAIN = 1
- LEAD/LAG = 2, GAIN = 1

LEAD = LAG

Der Funktionsbaustein verhält sich wie ein reiner Multiplikationsbaustein mit dem Multiplikator GAIN.

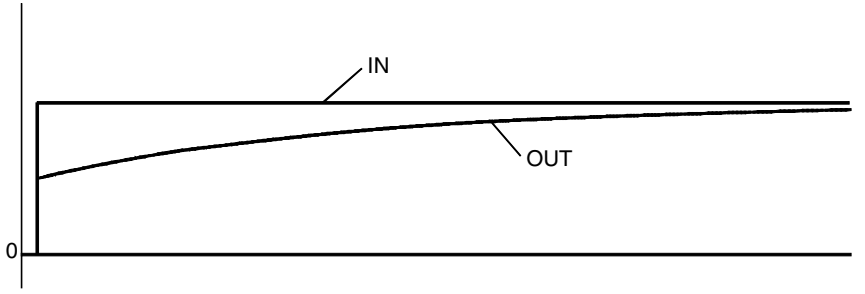
Funktionsbaustein LDLG mit LEAD = LAG



LEAD/LAG = 0.5, GAIN = 1

Der Ausgang OUT springt in diesem Fall auf die Hälfte des Endwertes um dann mit der Verzögerungszeitkonstante LAG in den Endwert (GAIN * IN) einzulaufen.

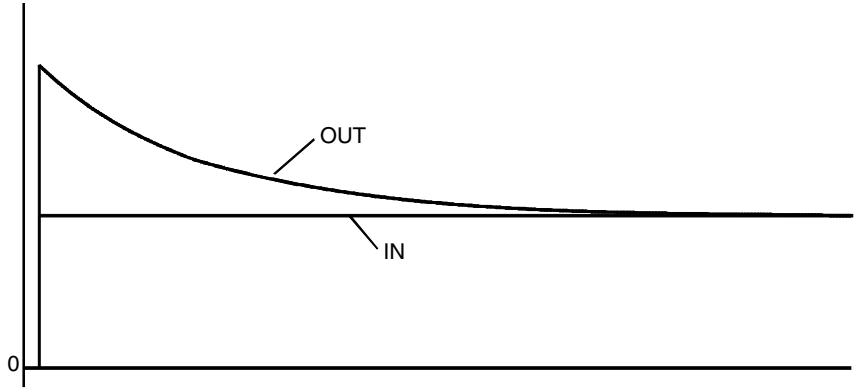
Funktionsbaustein LDLG mit LEAD/LAG = 0.5 und GAIN = 1



LEAD/LAG = 2, GAIN = 1

Der Ausgang OUT springt in diesem Fall auf das Doppelte des Endwertes um dann mit der Verzögerungszeitkonstante LAG in den Endwert (GAIN * IN) einzulaufen.

Funktionsbaustein LDLG mit LEAD/LAG = 2 und GAIN = 1



LEAD: Differenzierer mit Glättung

21

Übersicht

Einleitung

Dieses Kapitel beschreibt den Baustein LEAD .

Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Kurzbeschreibung	190
Darstellung	191
Detailbeschreibung	192

Kurzbeschreibung

Funktionsbeschreibung

Der Funktionsbaustein stellt ein Differenzierglied mit einem mit der Zeitkonstante LAG verzögerten Ausgang OUT dar.

Der Funktionsbaustein besitzt folgende Betriebsarten:

- Tracking
- Automatik

Als zusätzliche Parameter können EN und ENO projektiert werden.

Formel

Die Übertragungsfunktion für OUT lautet:

$$G(s) = GAIN \times \frac{s}{1 + s \times LAG}$$

Die Berechnungsformel lautet:

$$OUT = \frac{LAG}{dt + LAG} \times (OUT_{(old)} + GAIN \times (IN_{(new)} - IN_{(old)}))$$

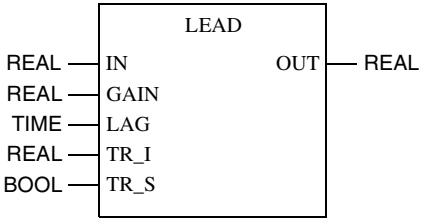
Bedeutung der Größen

Größe	Bedeutung
$IN_{(new)}$	Wert des Eingangs IN aus dem aktuellen Zyklus
$IN_{(old)}$	Wert des Eingangs IN aus dem vorherigen Zyklus
$OUT_{(old)}$	Wert des Ausgangs OUT aus dem vorherigen Zyklus
dt	Zeitdifferenz zwischen aktuellem Zyklus und vorherigem Zyklus

Darstellung

Symbol

Darstellung des Bausteins



Parameterbeschreibung

Beschreibung der Bausteinparameter

Parameter	Datentyp	Bedeutung
IN	REAL	Eingangswert
GAIN	REAL	Verstärkung der Differenzierung
LAG	TIME	Verzögerungszeitkonstante
TR_I	REAL	Initialisierungs-Eingang
TR_S	BOOL	Initialisierungs-Typ "1" = Betriebsart Tracking "0" = Betriebsart Automatik
OUT	REAL	Ausgang Differenzierer mit Glättung

Detailbeschreibung

Parametrierung

Die Parametrierung des Funktionsbausteins erfolgt durch Festlegen der Verstärkung GAIN des Differenzierers und der Zeitkonstante LAG, mit welcher der Ausgang OUT verzögert wird.

Für sehr kleine Abtastzeiten und dem Einheitssprung am Eingang IN (Sprung am Eingang IN von 0 auf 1.0) wird der Ausgang OUT auf den Wert GAIN springen (theoretischer Wert - real ein bißchen kleiner, bedingt durch die nicht unendlich kleine Abtastzeit), um dann mit der Zeitkonstante LAG wieder auf 0 zurückzugehen.

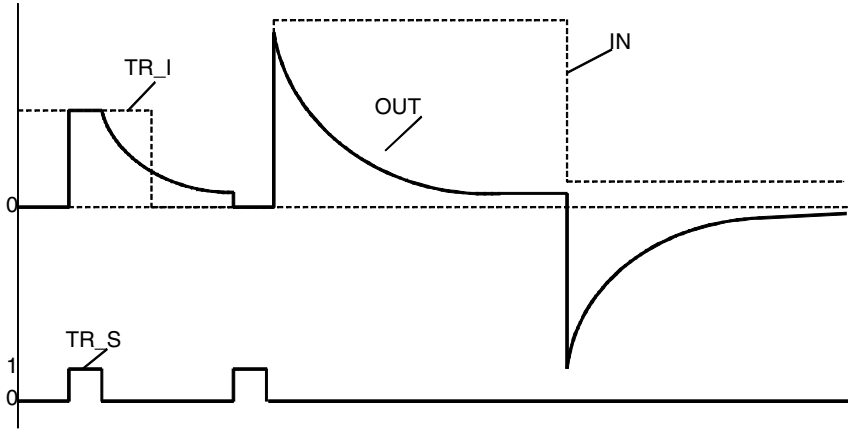
Betriebsarten

Es gibt zwei Betriebsarten, die über den Eingang TR_S ausgewählt werden:

Betriebsart	TR_S	Bedeutung
Automatik	0	Der Funktionsbaustein wird, wie in "Parametrierung" beschrieben, bearbeitet.
Tracking	1	Der Trackingwert TR_I wird fest auf den Ausgang OUT übertragen.

Beispiel

Darstellung der Sprungantwort des LEAD-Funktionsbausteins mit GAIN = 1 und LAG = 10s:



LEAD_LAG: PD-Glied mit Glättung

22

Übersicht

Einleitung

Dieses Kapitel beschreibt den Baustein LEAD_LAG.

Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Kurzbeschreibung	194
Darstellung	195
Detailbeschreibung	196
Beispiele zum Funktionsbaustein LEAD_LAG	197
Laufzeitfehler	199

Kurzbeschreibung

Funktionsbeschreibung

Der Funktionsbaustein stellt ein PD-Glied mit nachfolgendem Tiefpaß dar.

Der Funktionsbaustein besitzt folgende Eigenschaften:

- Definierbare Verzögerung des D-Anteils
- Betriebsarten Hand, Halt, Automatik

Als zusätzliche Parameter können EN und ENO projektiert werden.

Formel

Die Übertragungsfunktion lautet:

$$G(s) = gain \times \frac{1 + s \times lead}{1 + s \times lag}$$

Die Berechnungsformel lautet:

$$Y = \frac{lag \times Y_{(old)} + gain \times ((lead + dt) \times X - lead \times X_{(old)})}{lag + dt}$$

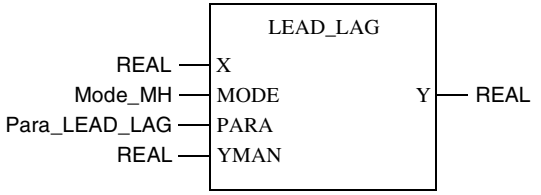
Bedeutung der Größen

Größe	Bedeutung
$X_{(old)}$	Wert des Eingangs X aus dem vorherigen Zyklus
$Y_{(old)}$	Wert des Ausgangs Y aus dem vorherigen Zyklus
dt	Zeitdifferenz zwischen aktuellem Zyklus und vorherigem Zyklus

Darstellung

Symbol

Darstellung des Bausteins



Parameterbeschreibung LEAD_LAG

Beschreibung der Bausteinparameter

Parameter	Datentyp	Bedeutung
X	REAL	Eingang
MODE	Mode_MH	Betriebsarten
PARA	Para_LEAD_LAG	Parameter
YMAN	REAL	Handwert-Stellwert
Y	REAL	Ausgang

Parameterbeschreibung Mode_MH

Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
man	BOOL	"1" = Betriebsart Hand
halt	BOOL	"1" = Betriebsart Halt

Parameterbeschreibung Para_LEAD_LAG

Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
gain	REAL	Verstärkungsfaktor
lead	TIME	Differenzierzeitkonstante
lag	TIME	Verzögerungszeitkonstante

Detailbeschreibung

Parametrierung

Die Parametrierung des Funktionsbausteins erfolgt durch Festlegen des Verstärkungsfaktors gain sowie der Parametrierung der Differenzierzeitkonstante lead und der Verzögerungszeitkonstante lag.

Für sehr kleine Abtastzeiten und dem Einheitssprung am Eingang X (Sprung am Eingang X von 0 auf 1.0) wird der Ausgang Y auf den Wert $\mathrm{gain} \times \mathrm{lead}/\mathrm{lag}$ springen (theoretischer Wert - real ein bißchen kleiner, bedingt durch die nicht unendlich kleine Abtastzeit), um sich dann mit der Zeitkonstante lag dem Wert $\mathrm{gain} \times 1.0$ zu nähern.

Betriebsarten

Es gibt drei Betriebsarten, die über die Elemente man und halt ausgewählt werden:

Betriebsart	man	halt	Bedeutung
Automatik	0	0	Der Funktionsbaustein wird, wie in "Parametrierung" beschrieben, bearbeitet.
Hand	1	0 oder 1	Der Handwert YMAN wird fest auf den Ausgang Y übertragen.
Halt	0	1	Der Ausgang Y wird auf dem zuletzt berechneten Wert gehalten. Der Ausgang wird nicht mehr verändert, kann jedoch vom Anwender überschrieben werden.

Beispiele zum Funktionsbaustein LEAD_LAG

Beispiel-Übersicht

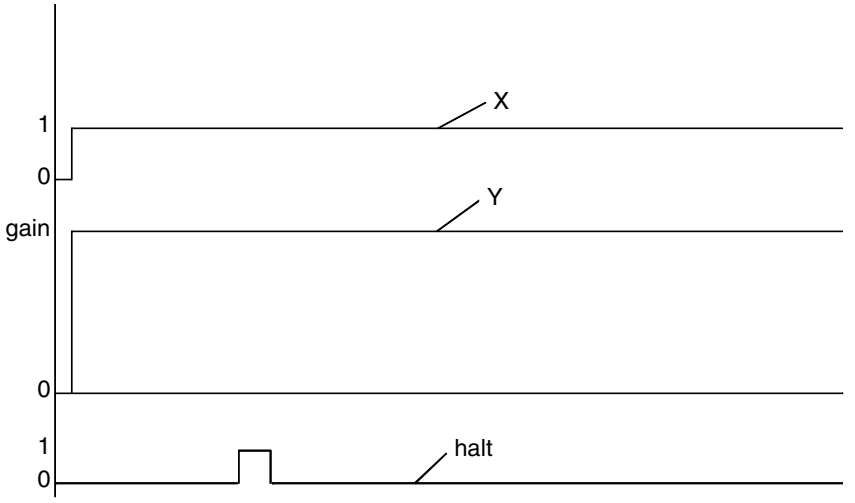
Folgende Beispiele werden in den nachfolgenden Diagrammen dargestellt:

- lead = lag
- lead=lag * 0.5, gain = 1
- lead/lag = 2, gain = 1

lead = lag

Der Funktionsbaustein verhält sich wie ein reiner Multiplikationsbaustein mit dem Multiplikator gain.

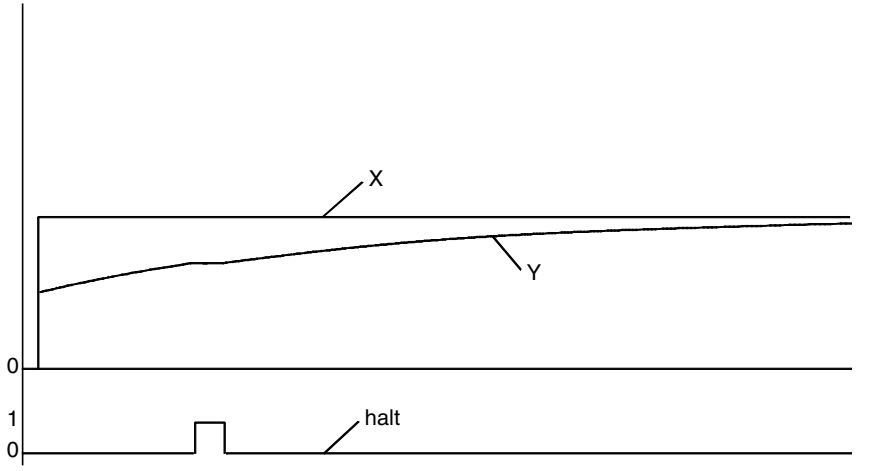
FunktionsbausteinLEAD_LAG mit lead = lag



lead=lag * 0.5, gain = 1

Der Ausgang Y springt in diesem Fall auf die Hälfte des Endwertes um dann mit der Verzögerungszeitkonstante lag in den Endwert (gain * X) einzulaufen.

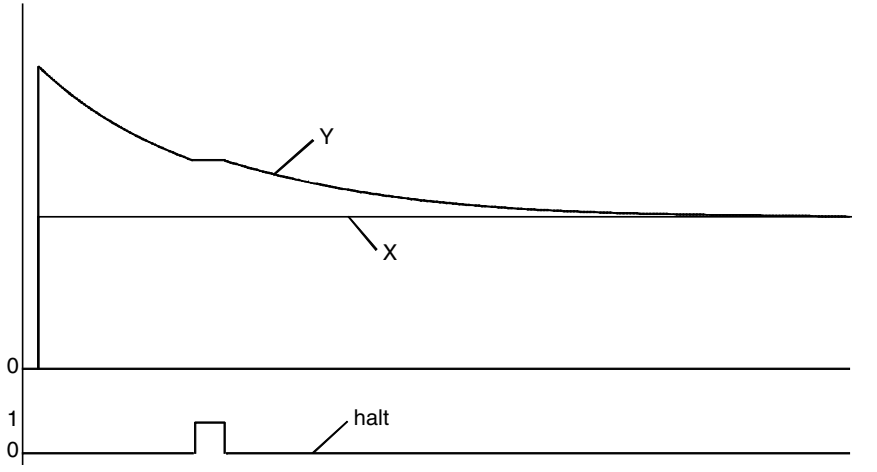
Funktionsbaustein LEAD_LAG mit lead/lag = 0.5 und gain = 1



lead/lag = 2, gain = 1

Der Ausgang Y springt in diesem Fall auf das Doppelte des Endwertes um dann mit der Verzögerungszeitkonstante lag in den Endwert (gain * X) einzulaufen.

Funktionsbaustein LEAD_LAG mit lead/lag = 2 und gain = 1



Laufzeitfehler

Fehlermeldung Es erfolgt eine Fehlermeldung, wenn eine unzulässige Gleitkommazahl am Eingang YMAN oder X anliegt.

LEAD_LAG1: PD-Glied mit Glättung

23

Übersicht

Einleitung

Dieses Kapitel beschreibt den Baustein LEAD_LAG1.

Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Kurzbeschreibung	202
Darstellung	203
Detailbeschreibung	204
Beispiele zum Funktionsbaustein LEAD_LAG1	205

Kurzbeschreibung

Funktionsbeschreibung

Der Funktionsbaustein stellt ein PD-Glied mit nachfolgender Glättung dar.

Der Funktionsbaustein besitzt folgende Eigenschaften:

- Definierbare Verzögerung des D-Anteils
- Betriebsarten Hand, Halt, Automatik

Als zusätzliche Parameter können EN und ENO projektiert werden.

Formel

Die Übertragungsfunktion lautet:

$$G(s) = GAIN \times \frac{1 + s \times LEAD}{1 + s \times LAG}$$

Die Berechnungsformel lautet:

$$Y = \frac{LAG \times Y_{(old)} + GAIN \times ((LEAD + dt) \times X - LEAD \times X_{(old)})}{LAG + dt}$$

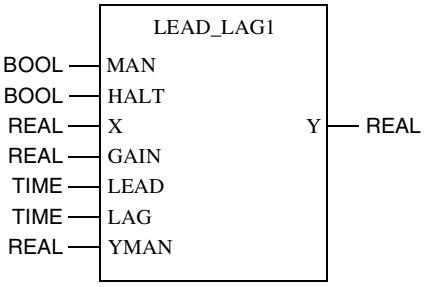
Bedeutung der Größen

Größe	Bedeutung
$X_{(old)}$	Wert des Eingangs X aus dem vorherigen Zyklus
$Y_{(old)}$	Wert des Ausgangs Y aus dem vorherigen Zyklus
dt	Zeitdifferenz zwischen aktuellem Zyklus und vorherigem Zyklus

Darstellung

Symbol

Darstellung des Bausteins



Parameterbeschreibung

Beschreibung der Bausteinparameter

Parameter	Datentyp	Bedeutung
MAN	BOOL	"1" = Betriebsart Hand
HALT	BOOL	"1" = Betriebsart Halt
X	REAL	Eingang
GAIN	REAL	Verstärkungsfaktor
LEAD	TIME	Differenzierzeitkonstante
LAG	TIME	Verzögerungszeitkonstante
YMAN	REAL	Handwert-Stellwert
Y	REAL	Ausgang

Detailbeschreibung

Parametrierung

Die Parametrierung des Funktionsbausteins erfolgt durch Festlegen des Verstärkungsfaktors GAIN sowie der Parametrierung der Differenzierzeitkonstante LEAD und der Verzögerungszeitkonstante LAG.

Für sehr kleine Abtastzeiten und dem Einheitssprung am Eingang X (Sprung am Eingang X von 0 auf 1.0) wird der Ausgang Y auf den Wert $GAIN \times LEAD/LAG$ springen (theoretischer Wert - real ein bißchen kleiner, bedingt durch die nicht unendlich kleine Abtastzeit), um dann mit der Zeitkonstante LAG auf den Wert $GAIN \times 1.0$ zuzugehen.

Betriebsarten

Es gibt drei Betriebsarten, die über die Eingänge MAN und HALT ausgewählt werden:

Betriebsart	MAN	HALT	Bedeutung
Automatik	0	0	Der Funktionsbaustein wird, wie in "Parametrierung" beschrieben, bearbeitet.
Hand	1	0 oder 1	Der Handwert YMAN wird fest auf den Ausgang Y übertragen.
Halt	0	1	Der Ausgang Y wird auf dem zuletzt berechneten Wert gehalten.

Beispiele zum Funktionsbaustein LEAD_LAG1

Beispiel-Übersicht

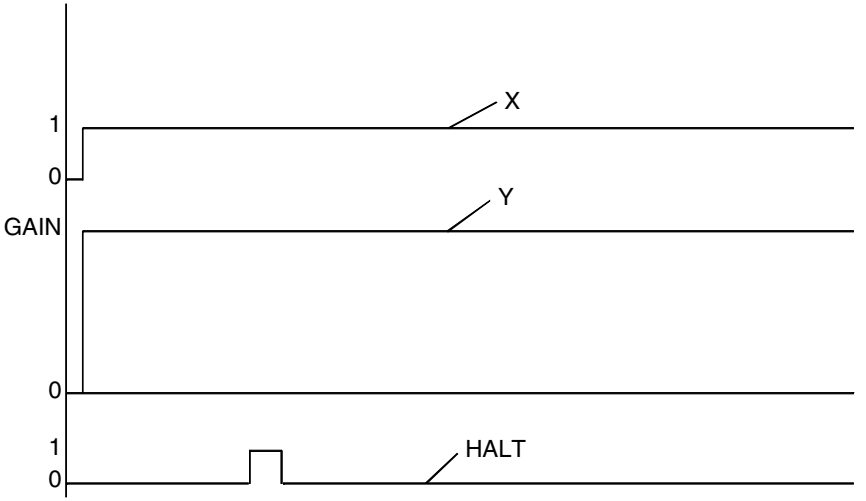
Folgende Beispiele werden in den nachfolgenden Diagrammen dargestellt:

- LEAD = LAG
- LEAD=LAG * 0.5, GAIN = 1
- LEAD/LAG = 2, GAIN = 1

LEAD = LAG

Der Funktionsbaustein verhält sich wie ein reiner Multiplikationsbaustein mit dem Multiplikator GAIN.

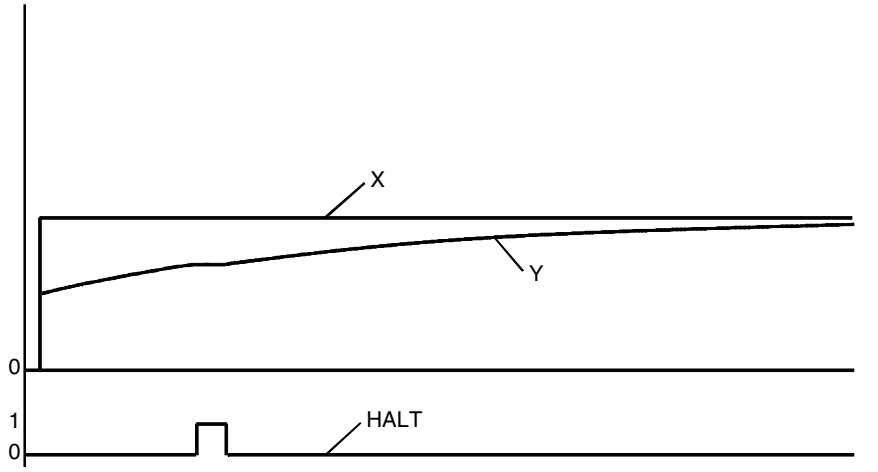
FunktionsbausteinLEAD_LAG1 mit LEAD = LAG



LEAD=LAG * 0.5, GAIN = 1

Der Ausgang Y springt in diesem Fall auf die Hälfte des Endwertes um dann mit der Verzögerungszeitkonstante lag in den Endwert (GAIN * X) einzulaufen.

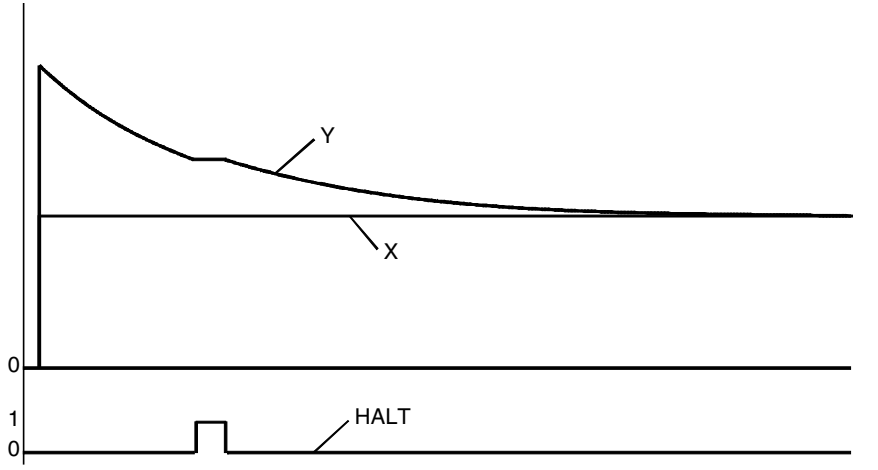
Funktionsbaustein LEAD_LAG1 mit LEAD/LAG = 0.5 und GAIN = 1



LEAD/LAG = 2, GAIN = 1

Der Ausgang Y springt in diesem Fall auf das Doppelte des Endwertes um dann mit der Verzögerungszeitkonstante LAG in den Endwert (GAIN * X) einzulaufen.

Funktionsbaustein LEAD_LAG1 mit LEAD/LAG = 2 und GAIN = 1



LIMV: Anstiegsbegrenzer 1. Ordnung

24

Übersicht

Einleitung

Dieses Kapitel beschreibt den Baustein LIMV.

Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Kurzbeschreibung	208
Darstellung	209
Detailbeschreibung	210
Laufzeitfehler	211

Kurzbeschreibung

Funktionsbeschreibung

Der Funktionsbaustein realisiert einen Anstiegsbegrenzer 1. Ordnung mit Begrenzung der Stellgröße.

Es wird der Gradient der Eingangsgröße X auf einen vorgebbaren Wert RATE begrenzt. Ferner wird der Ausgang Y durch YMAX und YMIN begrenzt. Dadurch kann der Funktionsbaustein Signale an die technologisch begrenzte Geschwindigkeit und Endlagen von Stellgliedern anpassen.

Als zusätzliche Parameter können EN und ENO projektiert werden.

Eigenschaften

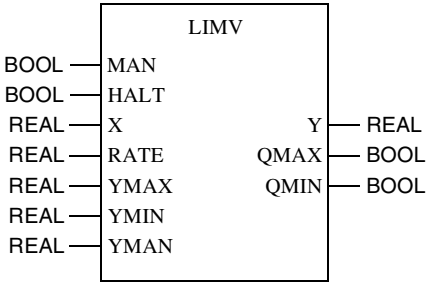
Der Funktionsbaustein besitzt folgende Eigenschaften:

- Betriebsarten Hand, Halt, Automatik
- Stellgrößen-Begrenzung im Automatik-Betrieb

Darstellung

Symbol

Darstellung des Bausteins



Parameterbeschreibung

Beschreibung der Bausteinparameter

Parameter	Datentyp	Bedeutung
MAN	BOOL	"1" = Betriebsart Hand
HALT	BOOL	"1" = Betriebsart Halt
X	REAL	Eingang
RATE	REAL	Maximale Anstiegsbegrenzung (maximales x')
YMAX	REAL	Obere Stellgrenze
YMIN	REAL	Untere Stellgrenze
YMAN	REAL	HandStellwert
Y	REAL	Ausgang
QMAX	BOOL	"1" = Ausgang Y, hat obere Grenze erreicht
QMIN	BOOL	"1" = Ausgang Y, hat untere Grenze erreicht

Detailbeschreibung

Parametrierung

Die Parametrierung des Funktionsbausteins erfolgt durch Festlegen der maximalen Anstiegsgeschwindigkeit RATE sowie der Grenzen YMAX und YMIN für den Ausgang Y. Die maximale Anstiegsgeschwindigkeit gibt an, um welchen Wert sich der Ausgang innerhalb einer Sekunde ändern darf.

Vom Parameter RATE wird der Betrag ausgewertet. Ist RATE = 0, wird Y = X.

Die Grenzen YMAX und YMIN begrenzen den Ausgang nach oben sowie nach unten. Damit gilt YMIN ≤ Y ≤ YMAX.

Das Erreichen der Grenzwerte, bzw. eine Begrenzung des Ausgangssignals wird durch die beiden Ausgänge QMAX und QMIN angezeigt:

- QMAX = 1 wenn Y ≥ YMAX
- QMIN = 1 wenn Y ≤ YMIN

Betriebsarten

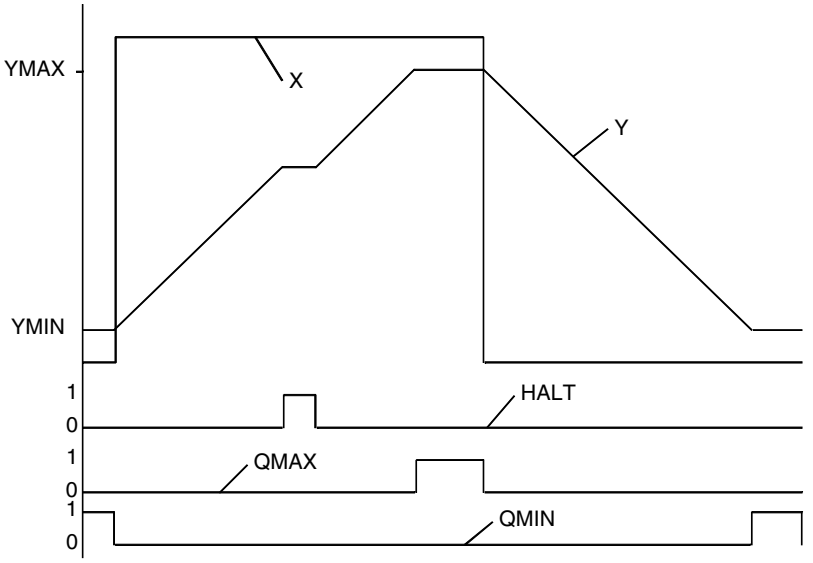
Es gibt drei Betriebsarten, die über die Eingänge MAN und HALT ausgewählt werden:

Betriebsart	MAN	HALT	Bedeutung
Automatik	0	0	Es wird ständig der aktuelle Wert für Y berechnet und ausgegeben.
Hand	1	0 oder 1	Der Handwert YMAN wird fest auf den Ausgang Y übertragen. Der Stellausgang wird jedoch durch YMAX und YMIN begrenzt.
Halt	0	1	Der Ausgang Y wird auf dem zuletzt berechneten Wert gehalten.

Beispiel

Dem Sprung am Eingang X folgt der Funktionsbaustein mit seiner maximalen Änderungsgeschwindigkeit. Während der Betriebsart Halt bleibt der Ausgang Y stehen, um anschließend an der Stelle weiterzufahren, an der er stehen geblieben ist. Deutlich zu erkennen ist auch die Begrenzung des Ausgangs Y durch YMAX und YMIN mit den dazugehörigen Meldungen QMAX und QMIN.

Dynamisches Verhalten von LIMV



Laufzeitfehler

Fehlermeldung

Bei YMAN < YMIN erfolgt eine Fehlermeldung.

MFLOW: Baustein für Massendurchfluß

25

Übersicht

Einleitung

Dieses Kapitel beschreibt den Baustein MFLOW.

Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Kurzbeschreibung	214
Darstellung	215
Detailbeschreibung	216
Laufzeitfehler	217

Kurzbeschreibung

Funktionsbeschreibung

Der Funktionsbaustein MFLOW berechnet den Massendurchfluß eines Gases in einem Drosselgerät aufgrund des Differentialdrucks und der Temperatur- und Druckbedingungen des Gases.

Der Meßwert des Differentialdrucks kann durch die Geschwindigkeit des Mediums oder einen anderen Meßwert mit Druck- und Temperaturkompensation ersetzt werden.

Als zusätzliche Parameter können EN und ENO projektiert werden.

Formel

Die vollständige Formel (d.h. mit en_sqrt = 1, en_pres = 1 und en_temp =1) lautet wie folgt:

$$\mathrm{OUT} = \mathrm{k} \times \sqrt{\frac{\mathrm{IN} \times \mathrm{PA}}{\mathrm{TA}}}$$

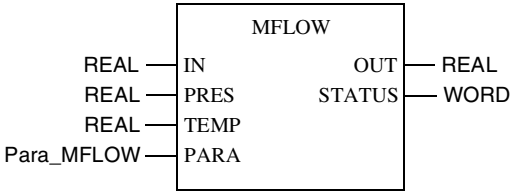
Bedeutung der Größen

Größe	Bedeutung
PA	Gasdruck in absoluten Einheiten
TA	Absolute Gastemperatur in Kelvin

Darstellung

Symbol

Darstellung des Bausteins



Parameterbeschreibung MFLOW

Beschreibung der Bausteinparameter

Parameter	Datentyp	Bedeutung
IN	REAL	Eingang
PRES	REAL	Absoluter oder relativer Gasdruck
TEMP	REAL	Gastemperatur ausgedrückt in °C oder °F
PARA	Para_MFLOW	Parameter
OUT	REAL	Wert des Massendurchflusses, mit Temperatur- und Druckkorrektur
STATUS	WORD	Statuswort

Parameterbeschreibung Para_MFLOW

Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
k	REAL	Rechenkonstante (siehe *Berechnung der Konstante k, S. 216*)
en_pres	BOOL	"1": Aktivierung der Druckkorrektur
pr_pa	BOOL	"1": PRES ist ein absoluter Druck "0": PRES ist ein relativer Druck
pu	REAL	Wert, der in der verwendeten Druckeinheit 1 Atmosphäre darstellt
en_temp	BOOL	"1": Aktivierung der Temperaturkorrektur
tc_tf	BOOL	"1": TEMP wird in Grad Fahrenheit ausgedrückt "0": TEMP wird in Grad Celsius ausgedrückt
en_sqrt	BOOL	"1": Berechnung mit Quadratwurzel

Detailbeschreibung

Berechnung der Konstante k

Die Konstante k kann aufgrund eines Referenz-Arbeitspunktes berechnet werden, bei dem der Massendurchfluß (MF_REF), der Differentialdruck (IN_REF), der absolute Druck (P_REF) und die absolute Temperatur (T_REF) bekannt sind.

Wenn der Eingang IN **ein Differentialdruck** ist, lautet die Formel wie folgt:

$$k = MF_REF \times \sqrt{\frac{T_REF}{P_REF \times IN_REF}}$$

Wenn der Eingang IN **kein Differentialdruck** ist, lautet die Formel wie folgt:

$$k = MF_REF$$

Spezifizierung der Berechnung

Bei der Berechnung ist eine einfache Multiplikation vorgegeben: $OUT = k \times IN$.
Um eine Druck- oder Temperaturkompensation zu erzielen, müssen die Parameter en_pres oder en_temp auf 1 eingestellt werden. Auch die Quadratwurzel ist nur aktiv, wenn en_sqrt = 1.

Wenn einer der Parameter en_sqrt, en_pres, en_temp auf 0 verbleibt, muß die Berechnung der Konstante k entsprechend angepaßt werden (Löschen der Quadratwurzel, Ersatz von P_REF oder T_REF durch 1)

Temperatur-einheit

Die Temperatur TEMP kann in Grad Celsius oder in Grad Fahrenheit ausgedrückt werden, je nach Wert des Parameters tc_tf :

tc_tf	Temperatureinheit von TEMP
0	Grad Celsius Berechnung der absoluten Temperatur TA: $TA(°K) = TEMP + 273$
1	Grad Fahrenheit Berechnung der absoluten Temperatur TA: $TA(°K) = \frac{5}{9} \times (TEMP - 32) + 273$

Druckeinheit

Der Druck PRES kann in einer beliebigen Einheit ausgedrückt werden, als absoluter oder relativer Druck, je nach Wert des Parameters pr_pa.

pr_pa	Druckeinheit von PRES
0	Relativer Druck Parameter pu muß in der verwendeten Einheit 1 Atmosphäre entsprechen. Berechnung des absoluten Drucks: PA = PRES + pu
1	Absoluter Druck: PA = PRES

Laufzeitfehler

Statuswort

Die Bits des Statuswortes haben folgende Bedeutung:

Bit	Bedeutung
Bit 0 = 1	Fehler in einer Berechnung in Gleitkommawerten
Bit 1 = 1	Erfassung eines unzulässigen Wertes an einem Gleitkommawerteingang
Bit 2 = 1	Division durch Null bei Berechnung in Gleitkommawerten
Bit 3 = 1	Kapazitätsüberlauf bei Berechnung in Gleitkommawerten
Bit 4 = 1	Eine der folgenden Größen ist negativ: IN, pu, PA, TA. Zur Berechnung verwendet der Funktionsbaustein den Wert 0.

Fehlermeldung

In folgenden Fällen wird ein Fehler gemeldet:

- An einem der Gleitkommawerteingänge wird ein unzulässiger Wert erfaßt
- Division durch Null bei Berechnung in Gleitkommawerten
- Kapazitätsüberlauf bei Berechnung in Gleitkommawerten

Der Ausgang OUT wird nicht verändert.

Warnung

Es wird eine Warnung ausgegeben, wenn der Parameter pu negativ ist, der Baustein kann in diesem Fall bei der Berechnung den Wert 0 anstelle des fehlerhaften Wertes pu verwenden.

MS: Handsteuerung eines Ausgangs

26

Übersicht

Einleitung

Dieses Kapitel beschreibt den Baustein MS.

Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Kurzbeschreibung	220
Darstellung	221
Detailbeschreibung	223
Beispiel	227
Laufzeitfehler	228

Kurzbeschreibung

Funktionsbeschreibung

Dieser Funktionsbaustein dient zur Steuerung eines numerischen Ausgangs, der einem über den Funktionsbaustein PWM1 (siehe *PWM1: Pulsbreitenmodulation, S. 421*) gesteuerten Analogausgang, Servomotor oder Stellglied aufgeschaltet werden kann. Die Steuerung kann über Bedienerdialog oder direkt über die SPS-Software erfolgen.

Im allgemeinen dient ein Regler-Funktionsbaustein zur Steuerung eines Digitalausgangs. Der MS-Baustein ist dann zu verwenden, wenn der Reglerausgang von der Steuerung des Analogausgangs entkoppelt werden soll.

Als zusätzliche Parameter können EN und ENO projektiert werden.

Anwendungsmöglichkeiten

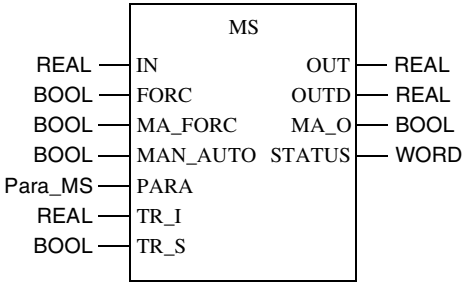
Der Funktionsbaustein wird hauptsächlich bei folgenden Anwendungen eingesetzt:

- Zur Steuerung eines Analogausgangs, der nicht über einen Regelkreis gesteuert wird (offener Regelkreis).
- Regelkreise, bei denen dem Reglerausgang und dem bedienergesteuerten Ausgang ein Verarbeitungsvorgang zwischengeschaltet ist.
- Bei Abtastung des den Ausgang steuernden Reglers, wenn die Abtastperiode 1 bis 2 Sekunden übersteigt.
- Bei Steuerung eines Servomotors: der Funktionsbaustein MS ist in diesem Fall dem Reglerbaustein und dem Servomotor zwischenzuschalten.

Darstellung

Symbol

Darstellung des Bausteins



Parameterbeschreibung MS

Beschreibung der Bausteinparameter

Parameter	Datentyp	Bedeutung
IN	REAL	In der Betriebsart Automatik verwendete Stellgröße
FORC	BOOL	"1": die Betriebsart Hand/Automatik wird über MA_FORC vorgegeben "0": die Betriebsart Hand/Automatik wird über MAN_AUTO vorgegeben
MA_FORC	BOOL	Betriebsart Hand/Auto (wenn FORC = 1) "1": Betriebsart Automatik "0": Betriebsart Hand
MAN_AUTO	BOOL	Betriebsart Hand/Automatik (wenn FORC = 0) "1": Betriebsart Automatik "0": Betriebsart Hand
PARA	Para_MS	Parameter
TR_I	REAL	Initialisierungseingang
TR_S	BOOL	Initialisierungsbefehl
OUT	REAL	Absoluter Ausgang
OUTD	REAL	Inkrementalausgang: Unterschied zwischen dem aktuellen Ausgang und dem Ausgang der vorherigen Ausführung
MA_O	BOOL	Aktuelle Betriebsart des Funktionsbausteins (0: Hand, 1: Automatik)
STATUS	WORD	Statuswort

Parameterbeschreibung Para_MS

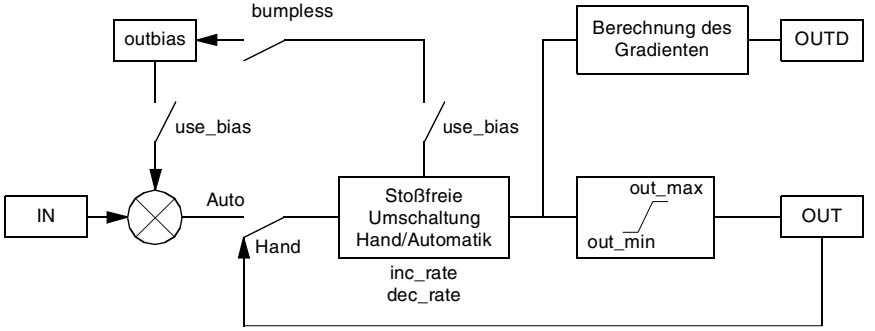
Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
out_min	REAL	unterer Grenzwert des Ausgangs
out_max	REAL	oberer Grenzwert des Ausgangs
inc_rate	REAL	Steigende Rampe bei Umschaltung Hand/ Automatik (Einheiten pro Sekunde)
dec_rate	REAL	Fallende Rampe bei Umschaltung Hand/Automatik (Einheiten pro Sekunde)
outbias	REAL	Wert des Bias
use_bias	BOOL	"1": Freigabe des Bias
bumpless	BOOL	"1": Einstellung des Bias bei Umschaltung Hand/ Automatik (stoßfrei)

Detailbeschreibung

Strukturbild

Im folgende Diagramm ist die Struktur des Funktionbausteines dargestellt:



Einstellung der Betriebsarten-auswahl

Die Betriebsartenauswahl kann in Abhängigkeit vom Eingang FORC entweder über das SPS-Programm oder über einen Bedienerdialog (Überwachungsgerät) eingestellt werden:

Eingang FORC	Einstellen der Betriebsart
0	Einstellung durch den Eingang MAN_AUTO (über Bediengerät): MAN_AUTO= 1: Betriebsart Automatik MAN_AUTO= 0: Betriebsart Hand In diesem Fall ist der Eingang MA_FORC unwirksam.
1	Einstellung durch den Eingang MA_FORC (über SPS-Programm): MA_FORC = 1: Betriebsart Automatik MA_FORC = 0: Betriebsart Hand In diesem Fall ist der Eingang MAN_AUTO unwirksam.

Der Ausgang MA_O gibt stets die aktuelle Betriebsart des Funktionsbausteins an.

Merkmale des Ausgangs OUT

Für den Ausgang OUT gelten folgende Merkmale:

- Betriebsart Automatik: Der Ausgang OUT ist die Kopie des Eingangs IN. In dieser Betriebsart kann dem Ausgang OUT ein OUTBIAS-Wert aufgeschaltet werden (use_bias auf 1 setzen). OUT berechnet sich nun wie folgt: OUT = IN + outbias.
- Betriebsart Hand: Der Funktionsbaustein setzt nicht den Ausgang, der Bediener kann den Wert der an den Ausgang OUT angeschlossenen Variable direkt ändern.
- Der Ausgang OUT ist grundsätzlich auf einen Bereich zwischen out_min und out_max begrenzt. Wenn der vom Funktionsbaustein berechnete (oder vom Bediener im Handmodus eingegebene) Wert einen dieser Grenzwerte überschreitet, wird der Wert von OUT gekappt (auf out_min oder out_max). Der Inkrementalausgang OUTD hingegen berücksichtigt diese Kappung nie.

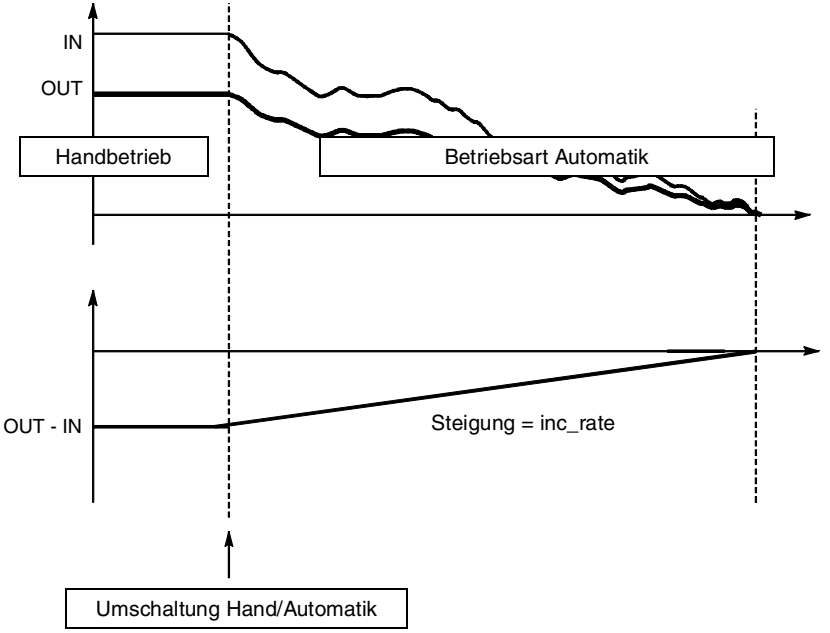
Umschaltung Hand/Automatik

Die Umschaltung Hand/Automatik am Ausgang erfolgt stoßfrei, da der Wert von IN nicht plötzlich auf den Ausgang geführt wird.

Der Ausgang OUT nähert sich dem Eingang IN rampenförmig mit positiver (inc_rate) oder negativer Steigung (dec_rate) an:

- inc_rate gilt, wenn IN zum Zeitpunkt der Umschaltung größer ist als OUT
- dec_rate gilt, wenn IN zum Zeitpunkt der Umschaltung kleiner ist als OUT

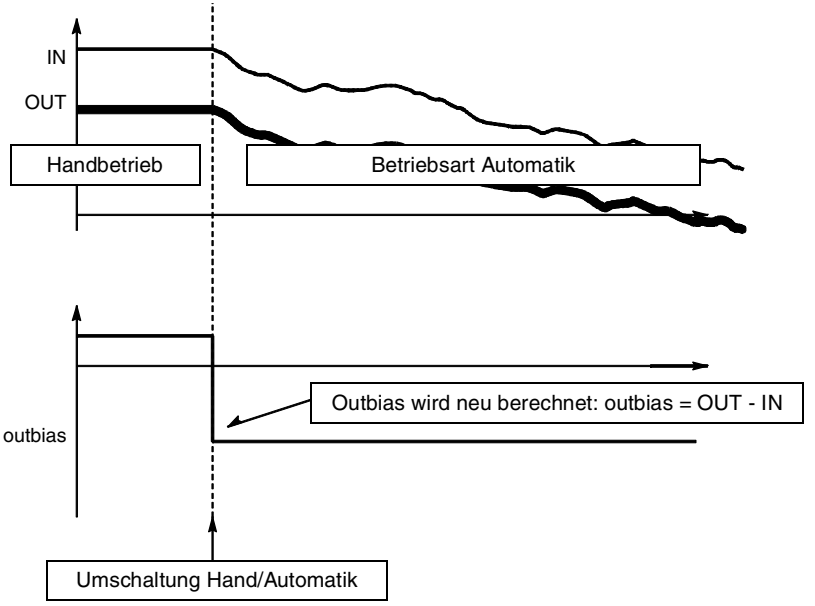
Stoßfreie Umschaltung



Die stoßfreie Umschaltung kann bei steigender Rampe annulliert werden, indem inc_rate auf 0 gesetzt wird. Ebenso ist bei dec_rate = 0 die Umschaltung bei fallender Rampe mit Stoß. In beiden Fällen wird der Eingang IN bei Umschaltung auf Automatikbetrieb sofort auf den Ausgang OUT geführt.

Wenn der Parameter outbias (use_bias = 1) verwendet wird, kann eine stoßfreie Umschaltung Hand/Automatik ohne Änderung des Ausgangs erzielt werden, indem der Parameter bumpless auf 1 gesetzt wird. Der Parameter outbias wird in diesem Fall vom Funktionsbaustein neu berechnet und berücksichtigt die Abweichung zwischen dem Eingang IN und dem Ausgang OUT.

Stoßfreie Umschaltung mit dem Parameter Outbias



Die stoßfreie Umschaltung Hand/Automatik ist sinnvoll, wenn der Eingang des Funktionsbausteins an keinen Regler oder an einen Reglerausgang ohne Integralanteil angeschlossen ist.

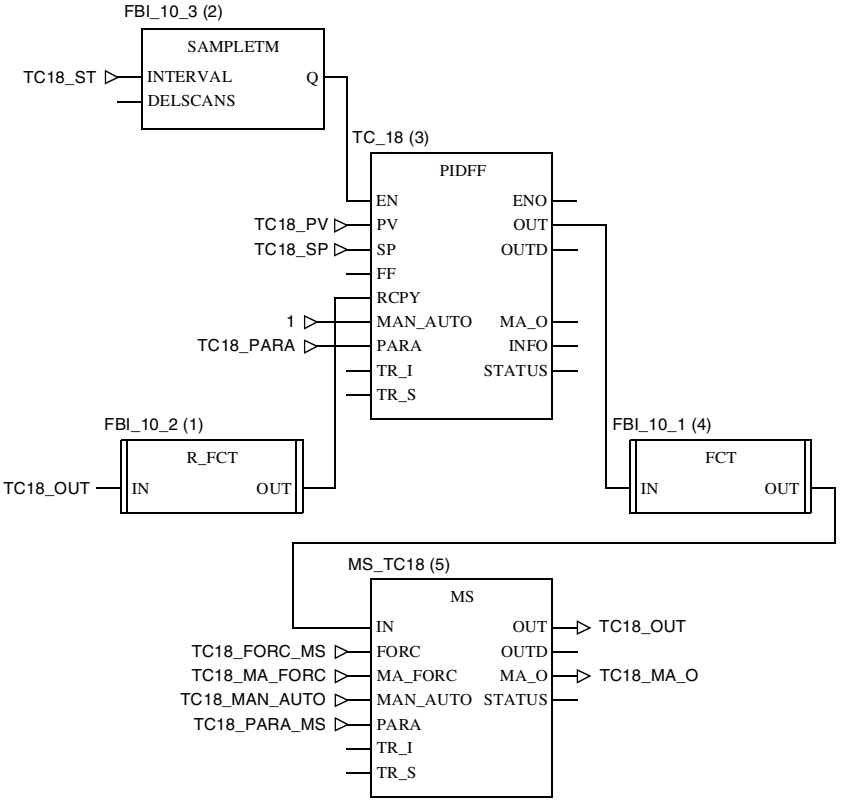
Beispiel

Beispiel

In diesem Beispiel wird dem Ausgang des Reglerbausteins und dem vom Bediener gesteuerten Ausgang ein Verarbeitungsvorgang (durch den DFB FCT) zwischengeschaltet.

Um eine stoßfreie Umschaltung zwischen den Betriebsarten Hand/Automatik zu gewährleisten, wird der umgekehrte Verarbeitungsvorgang (R_FCT) dem Ausgang des MS-Funktionsbausteins aufgeschaltet und das Ergebnis auf den Reglereingang RCPY zurückgeführt, der in der Betriebsart Automatik verbleibt (MAN_AUTO = 1).

Darstellung des Funktionsplanes



Laufzeitfehler

Statuswort

Die Bits des Statuswortes haben folgende Bedeutung:

Bit	Bedeutung
Bit 0 = 1	Fehler in einer Berechnung in Gleitkommawerten
Bit 1 = 1	Erfassung eines unzulässigen Wertes an einem der Gleitkommawerteingänge
Bit 2 = 1	Division durch Null bei einer Berechnung in Gleitkommawerten
Bit 3 = 1	Kapazitätsüberlauf bei einer Berechnung in Gleitkommawerten
Bit 4 = 1	Folgende Fehler werden angezeigt: ● Eine der folgenden Größen ist negativ: inc_rate, dec_rate. Zur Berechnung verwendet der Funktionsbaustein den Wert 0. ● Der Parameter Outbias liegt außerhalb des Bereichs $[(out_min - out_max), (out_max - out_min)]$. Der Funktionsbaustein verwendet in diesem Fall einen gekappten Wert: $(out_min - out_max)$ bzw. $(out_max - out_min)$.
Bit 5 = 1	Der Ausgang OUT hat den unteren Grenzwert out_min erreicht (siehe Hinweis)
Bit 6 = 1	Der Ausgang OUT hat den oberen Grenzwert out_max erreicht (siehe Hinweis)

Hinweis

Hinweis: Im Handbetrieb bleiben diese Bits nur einen Programmzyklus lang auf 1. Wenn der Anwender für OUT einen Wert eingibt, der einen der Grenzwerte überschreitet, setzt der Funktionsbaustein das Bit 5 oder 6 auf 1 und kappt den vom Bediener eingegebenen Wert. Bei der folgenden Ausführung des Funktionsbausteins liegt der Wert von OUT nicht mehr außerhalb des Bereichs und die Bits 5 und 6 werden wieder auf 0 gesetzt.

Fehlermeldung

Ein Fehler erscheint, wenn an einem Eingang ein nicht gleitender Wert erfaßt wird oder wenn bei einer Berechnung in Gleitkommawerten ein Problem auftritt. In diesem Fall bleiben die Ausgänge OUT, OUTD und MA_O unverändert.

Warnung

In folgenden Fällen wird eine Warnung ausgegeben::

- Der Parameter inc_rate ist negativ: der Funktionsbaustein verwendet in diesem Fall den Wert 0 anstelle des fehlerhaften Wertes von inc_rate.
- Der Parameter dec_rate ist negativ: der Funktionsbaustein verwendet in diesem Fall den Wert 0 anstelle des fehlerhaften Wertes von dec_rate.
- Der Parameter Outbias liegt außerhalb des Bereichs [(out_min - out_max), (out_max - out_min)]. Der Funktionsbaustein verwendet in diesem Fall bei der Berechnung den Wert (out_min - out_max) bzw. (out_max - out_min).

MULDIV_W: Multiplikation/ Division

27

Übersicht

Einleitung

Dieses Kapitel beschreibt den Baustein MULDIV_W.

Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Kurzbeschreibung	232
Darstellung	232
Laufzeitfehler	233

Kurzbeschreibung

Funktionsbeschreibung

Der Funktionsbaustein MULDIV_W führt eine gewichtete Multiplikation/Division von 3 numerischen Eingangsgrößen durch.

Als zusätzliche Parameter können EN und ENO projektiert werden.

Formel

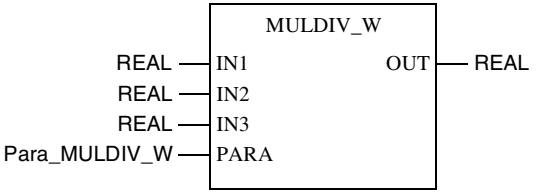
Die Formel lautet:

$$OUT = \frac{k \times (IN1 + c1) \times (IN2 + c2)}{IN3 + c3} + c4$$

Darstellung

Symbol

Darstellung des Bausteins



Parameterbeschreibung MULDIV_W

Beschreibung der Bausteinparameter

Parameter	Datentyp	Bedeutung
IN1 bis IN3	REAL	Zu verarbeitende numerische Größen
PARA	Para_MULDIV_W	Parameter
OUT	REAL	Ergebnis der Berechnung

Parameterbeschreibung PARA_ MULDIV_W

Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
k, c1 bis c4	REAL	Berechnungskoeffizienten

Laufzeitfehler

Fehlermeldung

Dieser Fehler wird gemeldet, wenn an einem Eingang ein nicht gleitender Wert erfaßt wird oder wenn in einer Berechnung in Gleitkommawerten ein Problem auftritt. Der Ausgang OUT behält im allgemeinen seinen vorherigen Wert bei, außer bei einer Division durch 0, wo der Wert INF entspricht, je nachdem, welches Vorzeichen der Zähler trägt.

PCON2: Zweipunkt-Regler

28

Übersicht

Einleitung

Diese Kapitel beschreibt den Baustein PCON2.

Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Kurzbeschreibung	236
Darstellung	237
Detailbeschreibung	238
Laufzeitfehler	240

Kurzbeschreibung

Funktionsbeschreibung

Der Funktionsbaustein bildet einen Zweipunktregler nach, der durch zwei dynamische Rückführungen PID-ähnliches Verhalten erhält.

Als zusätzliche Parameter können EN und ENO projektiert werden.

Eigenschaften

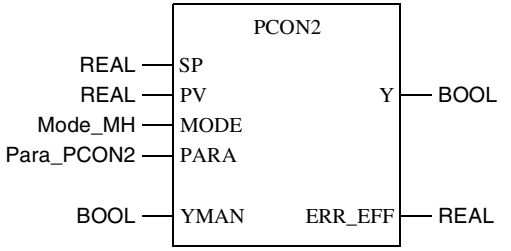
Der Funktionsbaustein besitzt folgende Eigenschaften:

- Betriebsarten Hand, Halt, Automatik
- zwei interne Rückführungen (Verzögerer 1. Ordnung)

Darstellung

Symbol

Darstellung des Bausteins:



Parameterbeschreibung PCON2

Beschreibung der Bausteinparameter

Parameter	Datentyp	Bedeutung
SP	REAL	Sollwerteingang
PV	REAL	Istwerteingang
MODE	Mode_MH	Betriebsarten
PARA	Para_PCON2	Parameter
YMAN	BOOL	"1" = Handwert für ERR_EFF
Y	BOOL	"1" = Ausgang Stellgröße
ERR_EFF	REAL	Wirksamer Schaltwert

Parameterbeschreibung Mode_MH

Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
man	BOOL	"1"= Betriebsart Hand
halt	BOOL	"1"= Betriebsart Halt

Parameterbeschreibung Para_PCON2

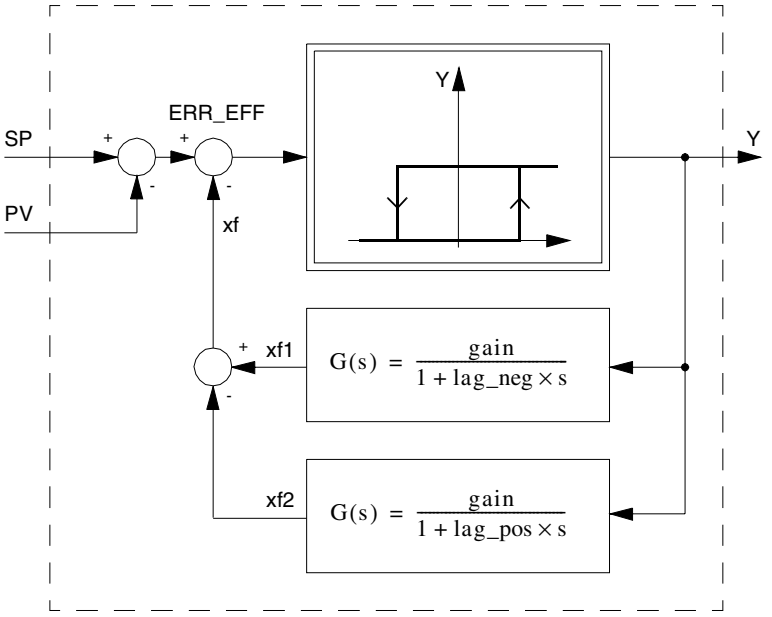
Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
gain	REAL	Rückführverstärkung
lag_neg	TIME	Zeitkonstante der schnellen Rückführung
lag_pos	TIME	Zeitkonstante der langsamen Rückführung
hys	REAL	Hysterese vom Zweipunktschalter
xf_man	REAL	Rücksetzwert der Rückführung in % (0 - 100)

Detailbeschreibung

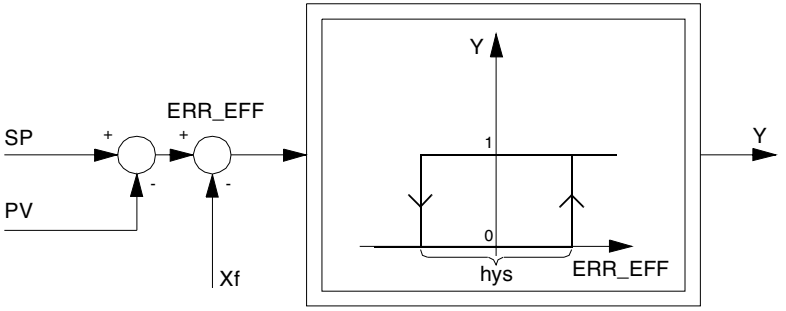
Struktur des Reglers

Struktur des Zweipunktreglers:



Prinzip des Zweipunkt-reglers

Dem eigentlichen Zweipunktregler werden 2 dynamische Rückführungen (PT1-Glieder) hinzugefügt. Durch geeignete Wahl der Zeitkonstanten dieser Rückführglieder erhält der Zweipunktregler ein dynamisches Verhalten, daß dem Verhalten eines PID-Reglers entspricht.



Rückführung

Der Rückführparametersatz, bestehend aus der Rückführverstärkung gain und den Rückführzeitkonstanten lag_neg und lag_pos, ermöglicht einen universellen Einsatz des Zweipunktreglers.

Folgende Tabelle gibt dazu genaueren Aufschluß:

Rückführung	lag_neg	lag_pos
2-Punkt-Verhalten (ohne Rückführung)	= 0	= 0
negative Rückführung	> 0	= 0
negative + positive Rückführung	> 0	> lag_neg
Warnung, da Mitkopplung (neg. Feedback mit lag_pos)	= 0	> 0
Warnung, da Mitkopplung (pos. Feedback abgeschaltet)	> lag_pos	> 0

Die Rückführverstärkung gain ist größer Null zu wählen!

Für xf_man (Bedeutung 0% bis 100%) sind Werte zwischen 0 und 100 einzutragen!

Hysterese

Der Parameter hys gibt die Schalthysterese an, d.h. den Wert, um den sich der wirksame Schaltwert ERR_EFF ausgehend vom Schaltpunkt hys/2 verkleinern muß bevor der Ausgang Y wieder auf "0" gesetzt wird. Die Abhängigkeit des Ausgangs Y abhängig von dem wirksamen Schaltwert ERR_EFF und dem Parameter hys wird im Bild *Prinzip des Zweipunktreglers, S. 238* deutlich. Der Wert des Parameters hys wird typisch auf 1% des maximalen Regelbereich [max. (SP - PV)] eingestellt.

Betriebsarten

Es gibt drei Betriebsarten, die über die Elemente man und halt ausgewählt werden:

Betriebsart	man	halt	Bedeutung
Automatik	0	0	Der Funktionsbaustein wird, wie oben beschrieben, bearbeitet.
Hand	1	0 oder 1	Der Ausgang Y wird auf den Wert YMAN gesetzt. xf1 und xf2 werden nach folgender Formel berechnet: xf1 = xf_man * gain /100 xf2 = xf_man * gain /100
Halt	0	1	Der Ausgang Y wird auf dem letzten Wert festgehalten. xf1 und xf2 werden auf gain * Y gesetzt.

Laufzeitfehler

Warnung

In folgenden Fällen wird eine Warnung ausgegeben:

Ursachen	Verhalten des Reglers
lag_neg = 0 und lag_pos > 0	Der Regler arbeitet, als hätte er nur eine negative Rückführung mit der Zeitkonstanten lag_pos.
lag_pos < lag_neg > 0	Der Regler arbeitet, als hätte er nur eine negative Rückführung mit der Zeitkonstanten lag_neg.
xf_man < 0 oder xf_man > 100	Der Regler arbeitet ohne interne Rückführungen.

PCON3: Dreipunkt-Regler

29

Übersicht

Einleitung

Diese Kapitel beschreibt den Baustein PCON3.

Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Kurzbeschreibung	242
Darstellung	243
Detailbeschreibung	245
Laufzeitfehler	247

Kurzbeschreibung

Funktionsbeschreibung

Der Funktionsbaustein bildet einen Dreipunktregler nach, der durch zwei dynamische Rückführungen PID-ähnliches Verhalten erhält.

Als zusätzliche Parameter können EN und ENO projektiert werden.

Eigenschaften

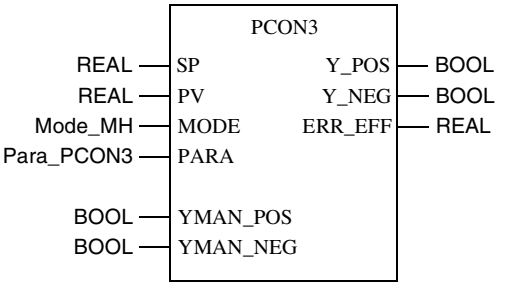
Der Funktionsbaustein PCON3 besitzt folgende Eigenschaften:

- Betriebsarten Hand, Halt, Automatik
- zwei interne Rückführungen (Verzögerer 1. Ordnung)

Darstellung

Symbol

Darstellung des Bausteins:



Parameterbeschreibung PCON3

Beschreibung der Bausteinparameter

Parameter	Datentyp	Bedeutung
SP	REAL	Sollwerteingang
PV	REAL	Istwerteingang
MODE	Mode_MH	Betriebsarten
PARA	Para_PCON3	Parameter
YMAN_POS	BOOL	Hand-Stellwert für Y_POS
YMAN_NEG	BOOL	Hand-Stellwert für Y_NEG
Y_POS	BOOL	"1" = positive Stellgröße an Ausgang ERR_EFF
Y_NEG	BOOL	"1" = negative Stellgröße an Ausgang ERR_EFF
ERR_EFF	REAL	Wirksamer Schaltwert

Parameterbeschreibung Mode_MH

Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
man	BOOL	"1"= Betriebsart Hand
halt	BOOL	"1"= Betriebsart Halt

Parameterbeschreibung Para_PCON3

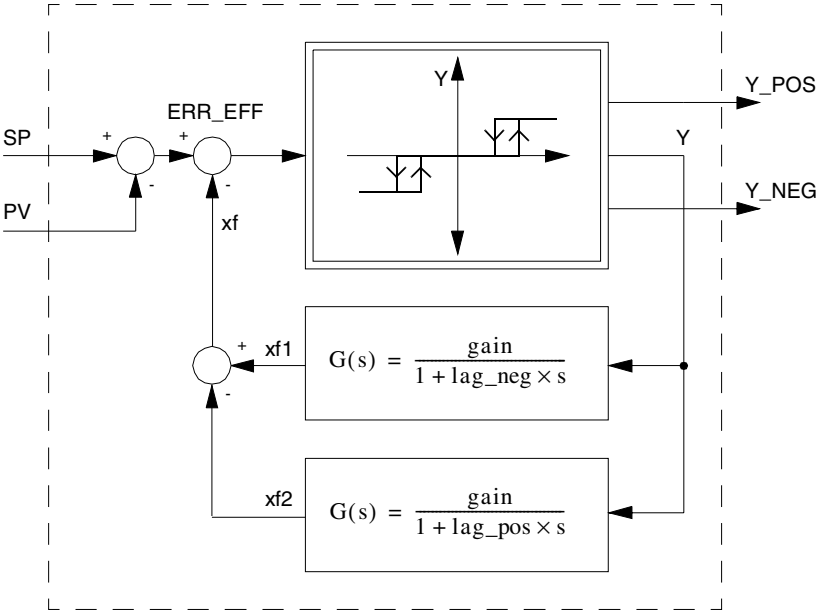
Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
gain	REAL	Rückführverstärkung (Rückführ-Parameter-Satz)
lag_neg	TIME	Zeitkonstante der schnellen Rückführung (Rückführ-Parameter-Satz)
lag_pos	TIME	Zeitkonstante der langsamen Rückführung (Rückführ-Parameter-Satz)
hys	REAL	Hysterese vom Dreipunktschalter
db	REAL	Unempfindlichkeitszone
xf_man	REAL	Rücksetzwert der Rückführung in % (0 - 100)

Detailbeschreibung

Struktur des Reglers

Struktur des Dreipunktreglers:

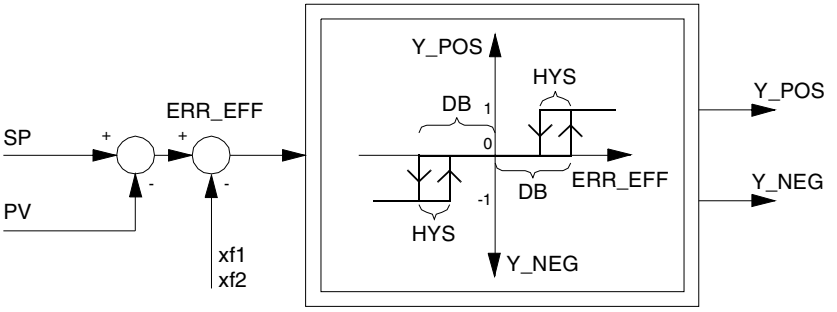


Dabei gilt:

Wenn...	Dann...
Y = 1	Y_POS = 1 Y_NEG = 0
Y = 0	Y_POS = 0 Y_NEG = 0
Y = -1	Y_POS = 0 Y_NEG = 1

Prinzip des Dreipunktreglers

Dem eigentlichen Dreipunktregler werden 2 dynamische Rückführungen (PT1-Glieder) hinzugefügt. Durch geeignete Wahl der Zeitkonstanten dieser Rückführglieder erhält derDreipunktregler ein dynamisches Verhalten, daß dem Verhalten eines PID-Reglers entspricht.



Rückführung

Der Funktionsbaustein hat einen Parametersatz für die internen Rückführungen, bestehend aus der Rückführverstärkung gain und den Rückführzeitkonstanten lag_neg und lag_pos.

Folgende Tabelle gibt dazu genaueren Aufschluß:

Rückführung	**lag_neg**	**lag_pos**
3-Punkt-Verhalten (ohne Rückführung)	= 0	= 0
negative Rückführung	> 0	= 0
negative + positive Rückführung	> 0	> lag_neg
Warnung, da Mitkopplung (neg. Feedback mit lag_pos)	= 0	> 0
Warnung, da Mitkopplung (pos. Feedback abgeschaltet)	> lag_pos	> 0

Der Parameter gain muß > 0 sein.

Von der Hysterese hys und der Unempfindlichkeitszone db wird der Betrag ausgewertet!

Für xf_man (Bedeutung -100 bis 100%) sind Werte zwischen -100 und 100 einzutragen!

Unempfindlichkeitszone

Der Parameter db legt den Einschaltpunkt für die Ausgänge Y_POS und Y_NEG fest. Ist der wirksame Schaltwert ERR_EFF positiv und wird größer als db, dann schaltet der Ausgang Y_POS von "0" auf "1". Ist wirksame Schaltwert ERR_EFF negativ und wird kleiner als -db, dann schaltet der Ausgang Y_NEG von "0" auf "1". Der Wert des Parameters db wird typisch auf 1% des maximalen Regelbereiches (max. SP - PV) eingestellt.

Hysterese

Der Parameter hys gibt die Schalthysterese an, d.h. den Wert, um den sich der wirksame Schaltwert ERR_EFF ausgehend vom Schaltpunkt db verkleinern muß bevor der Ausgang Y_POS (Y_NEG) wieder auf "0" gesetzt wird. Der Zusammenhang zwischen Y_POS und Y_NEG abhängig vom wirksamen Schaltwert ERR_EFF und den Parametern db und hys wird im Bild *Prinzip des Dreipunktreglers, S. 246* deutlich. Der Wert des Parameters hys wird typisch auf 0.5% des maximalen Regelbereiches (max. SP - PV) eingestellt.

Betriebsarten

Es gibt drei Betriebsarten, die über die Elemente man und halt ausgewählt werden:

Betriebsart	man	halt	Bedeutung
Automatik	0	0	Der Funktionsbaustein wird, wie oben beschrieben, bearbeitet.
Hand	1	0 oder 1	Die Ausgänge Y_POS und Y_NEG werden auf den Wert YMAN_POS und YMAN_NEG gesetzt. Hierbei ist die Vorranglogik - Y_NEG ist dominant über Y_POS - eingebaut, welche verhindert, daß beide Ausgänge gleichzeitig gesetzt sein können. xf1 und xf2 werden nach folgender Formel berechnet: xf1 = xf_man * gain /100 xf2 = xf_man * gain /100
Halt	0	1	In der Betriebsart Halt werden beide Ausgänge Y_POS und Y_NEG auf dem jeweils letzten Wert festgehalten. xf1 und xf2 werden auf gain * Y gesetzt.

Laufzeitfehler

Fehlermeldung

Bei hys > 2 * db erfolgt eine Fehlermeldung.

Warnung

In folgenden Fällen wird eine Warnung ausgegeben:

Ursachen	Verhalten des Reglers
lag_neg = 0 und lag_pos > 0	Der Regler arbeitet, als hätte er nur eine negative Rückführung mit der Zeitkonstanten lag_pos.
lag_pos < lag_neg > 0	Der Regler arbeitet, als hätte er nur eine negative Rückführung mit der Zeitkonstanten lag_neg.
xf_man < 0 oder xf_man > 100	Der Regler arbeitet ohne interne Rückführungen.

PD_or_PI: Strukturumschaltung PD- /PI-Regler

30

Übersicht

Einleitung

Diese Kapitel beschreibt den Baustein PD_or_PI.

Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Kurzbeschreibung	250
Darstellung	251
Strukturbild des Funktionsbausteins PD_or_PI	253
Detailbeschreibung	254
Detailierte Formeln	257
Laufzeitfehler	259

Kurzbeschreibung

Funktionsbeschreibung

Der Funktionsbaustein PD_or_PI kann sowohl als PD-Regler als auch als PI-Regler arbeiten. Abhängig von der Regeldifferenz (SP - PV) und einem vorgebbaren Schaltwert trig_err wird automatisch eine Strukturumschaltung von PD- zum PI-Regler und umgekehrt von PI- zum PD-Regler durchgeführt.

Dieser EFB eignet sich speziell für Anfahrschaltungen. Der Regler reagiert beim Hochfahren eines Prozesses als P(D)-Regler, wobei die Regelgröße möglichst schnell den eingestellten Wert der Führungsgröße erreichen soll. Ist der vorgegebene Sollwert fast erreicht, so wird die Reglerstruktur umgeschaltet und ein I-Anteil sorgt nun dafür, daß die bleibende Regelabweichung verschwindet.

Als zusätzliche Parameter können EN und ENO projektiert werden.

Eigenschaften

Der Funktionsbaustein besitzt folgende Eigenschaften:

- PI-Regler mit unabhängiger gain, ti-Einstellung
- PD-Regler mit unabhängiger gain, td-Einstellung
- Stellgrößen-Begrenzung im Automatik-Betrieb
- Antiwindup-Reset bei PI-Betrieb
- definierbare Verzögerung des D-Anteils
- Betriebsarten Hand, Halt, Automatik
- stoßfreie Umschaltung Hand/Automatik
- automatische stoßfreie Umschaltung von PD-Betrieb auf PI-Betrieb und zurück

Übertragungsfunktion des PI-Reglers

Die Übertragungsfunktion des PI-Reglers lautet:

$$G(s) = gain_i \times \left(1 + \frac{1}{ti \times s}\right)$$

Übertragungsfunktion des PD-Reglers

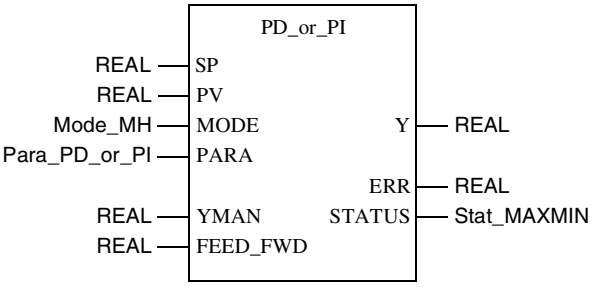
Die Übertragungsfunktion des PD-Reglers lautet:

$$G(s) = gain_d \times \left(1 + \frac{td \times s}{1 + td_lag \times s}\right)$$

Darstellung

Symbol

Darstellung des Bausteins:



Parameterbeschreibung PD_or_PI

Beschreibung der Bausteinparameter

Parameter	Datentyp	Bedeutung
SP	REAL	Sollwerteingang (Führungsgröße)
PV	REAL	Istwerteingang (Regelgröße)
MODE	Mode_MH	Betriebsart
PARA	Para_PD_or_PI	Parameter
YMAN	REAL	Handwert Stellgröße
FEED_FWD	REAL	Störgröße
Y	REAL	Stellgröße
ERR	REAL	Regeldifferenz
STATUS	Stat_MAXMIN	Ausgangs-Status

Parameterbeschreibung Mode_MH

Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
man	BOOL	"1": Betriebsart Hand
halt	BOOL	"1": Betriebsart Halt

Parameterbeschreibung Para_PD_or_PI

Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
trig_err	REAL	Schaltwert für Umschaltung von PD- zum PI-Regler
gain_d	REAL	Proportionalbeiwert (Verstärkungsfaktor) des PD-Reglers
td	TIME	Vorhaltezeit des PD-Reglers
td_lag	TIME	Verzögerung der Vorhaltezeit des PD-Reglers
gain_i	REAL	Proportionalbeiwert (Verstärkungsfaktor) des PI-Reglers
ti	TIME	Nachstellzeit des PI-Reglers
ymax	REAL	Obere Stellgrenze
ymin	REAL	Untere Stellgrenze

Parameterbeschreibung Stat_MAXMIN

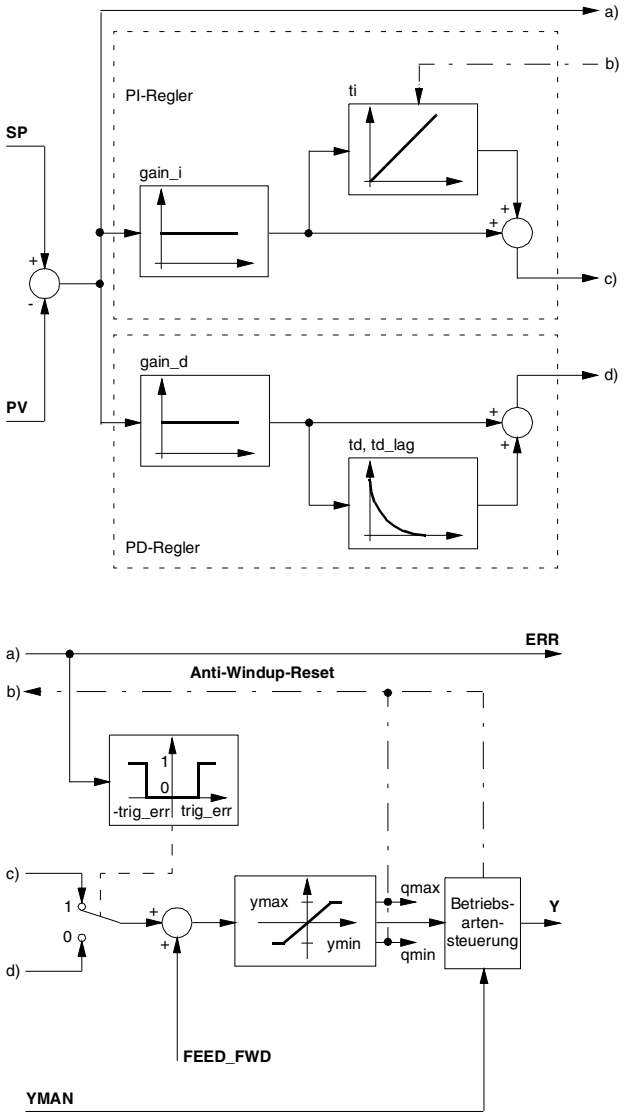
Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
qmax	BOOL	"1" = Y hat obere Stellgrenze erreicht
qmin	BOOL	"1" = Y hat untere Stellgrenze erreicht

Strukturbild des Funktionsbausteins PD_or_PI

Strukturbild

Nachfolgend finden Sie das Strukturbild des Bausteins PD_or_PI:



Detailbeschreibung

Festlegung des Schaltwertes

Die Parametrierung beginnt mit der Festlegung des Schaltwertes trig_err. Über diesen Parameter wird festgelegt, wann der Funktionsbaustein vom PD-Betrieb automatisch in den PI-Betrieb wechseln soll.

Wird die Regeldifferenz ERR = SP - PV betragsmäßig kleiner als der Schaltwert trig_err, so schaltet der Regler automatisch von PD-Betrieb auf PI-Betrieb um.

Wird die Regeldifferenz ERR betragsmäßig größer als der Schaltwert trig_err, so schaltet der Regler automatisch von PI-Betrieb auf PD-Betrieb um.

Daraus ergibt sich:
- PD-Regler: ERR > trig_err
- PI-Regler: ERR ≤ trig_err

Zu jedem Reglertyp gehört ein eigener Parametersatz, der ebenfalls projektiert werden muß. Mit der Umschaltung der Reglerstruktur wird praktisch auch die Umschaltung von einem Parametersatz auf den anderen Parametersatz vollzogen. Die Umschaltung geschieht stoßfrei.

PD-Regler

Die Parametrierung des PD-Reglers geschieht durch Projektierung des Proportionalbeiwertes gain_d und der Vorhaltezeit td.

Für den Betrieb als PD-Regler wird der D-Anteil mit der Zeitkonstanten td_lag verzögert. Das Verhältnis von td/td_lag nennt man Differenzierverstärkung und wird im allgemeinen zwischen 3 und 10 gewählt. Der D-Anteil wird basierend auf der Regeldifferenz ERR bestimmt, so daß bei Führungsgrößenänderungen (Änderungen im Eingang SP) ein Sprung bedingt durch den D-Anteil entsteht.

Der D-Anteil kann mit td = 0 abgeschaltet werden.

PI-Regler

Die Parametrierung des PI-Reglers geschieht durch Projektierung des Proportionalbeiwertes gain_i und der Nachstellzeit ti.

Im allgemeinen wird der Proportionalbeiwert während des Anfahrvorganges mit PD-Struktur höher eingestellt als im anschließenden annähernd stationären Betrieb mit PI-Struktur. Diesem Umstand ist durch Vergabe zweier unabhängiger Proportionalbeiwerte Rechnung getragen.

Der I-Anteil kann mit ti = 0 abgeschaltet werden.

Stellgrößen-Begrenzung

Die Grenzen ymax und ymin begrenzen die Stellgröße nach oben sowie nach unten.

Damit gilt: ymin ≤ Y ≤ ymax

Das Erreichen der Grenzwerte bzw. eine Begrenzung der Stellgröße wird durch qmax und qmin angezeigt:

- qmax = 1, wenn Y ≥ ymax
- qmin = 1, wenn Y ≤ ymin

Die Obergrenze ymax für die Begrenzung der Stellgröße ist größer als die Untergrenze ymin zu wählen.

Antiwindup-Reset

Findet eine Stellgrößen-Begrenzung statt und ist die aktuelle Reglerstruktur PI, so soll durch Antiwindup-Reset dafür gesorgt werden, daß der integrale Anteil nicht "über alle Grenzen" wachsen kann. Die Antiwindup-Maßnahme wird nur dann durchgeführt, wenn der I-Anteil des Reglers nicht "0" ist. Die Grenzen für Antiwindup sind hier dieselben, wie für die Stellgrößenbegrenzung.

Die Antiwindup-Reset-Maßnahme korrigiert den I-Anteil in der Form, daß gilt:

- YI ≥ ymin - gain_i * (SP - PV) - FEED_FWD
- YI ≤ ymax - gain_i * (SP - PV) - FEED_FWD

Betriebsarten

Es gibt drei Betriebsarten, die über die Elemente man und halt ausgewählt werden:

Betriebsart	man	halt	Bedeutung
Automatik	0	0	Die Stellgröße Y wird durch den diskretisierten PI- oder PD-Algorithmus, in Abhängigkeit von der Regelgröße PV und der Führungsgröße SP bestimmt. Die Stellgröße wird durch ymax und ymin begrenzt. Die Stellgrenzen sind gleichzeitig auch Grenzen für den Antiwindup-Reset.
Hand	1	0 oder 1	Der Handstellwert YMAN wird fest auf die Stellgröße Y übertragen. Die Stellgröße wird durch ymax und ymin begrenzt. Die internen Größen werden so nachgeführt, daß der Regler stoßfrei von Hand auf Automatik umgeschaltet werden kann.
Halt	0	1	Der Reglerausgang bleibt, wie er vorgefunden wird, d.h. der Baustein verändert die Stellgröße Y nicht (Regler bleibt stehen). Die internen Größen werden so nachgeführt, daß der Regler ohne Stoß von seiner aktuellen Lage aus weiterfährt. Stellgrößenbegrenzung und Antiwindup-Maßnahme sind wie im Automatik-Betrieb. Die Betriebsart Halt eignet sich unter anderem auch dafür, die Stellgröße Y über ein externes Bediengerät zu stellen, wobei die internen Anteile im Regler richtig nachgeführt werden.

Detailierte Formeln

Erläuterung der Formelgrößen

Bedeutung der Größen in den nachfolgenden Formeln:

Größe	Bedeutung
dt	Aktuelle Abtastzeit
ERR	Regeldifferenz
$ERR_{(old)}$	Wert der Regeldifferenz aus dem vorherigen Abtastschritt
FEED_FWD	Störgröße
Y	Aktueller Ausgang (Betriebsart Halt) oder YMAN (Betriebsart Hand)
YD	D-Anteil
$YD_{(old)}$	Wert des D-Anteils aus dem vorherigen Abtastschritt
YI	I-Anteil
$YI_{(old)}$	Wert des I-Anteils aus dem vorherigen Abtastschritt.
YP	P-Anteil

Regeldifferenz

Die Regeldifferenz wird wie folgt ermittelt:

$$ERR = SP - PV$$

Stellgröße

Die Stellgröße setzt sich aus verschiedenen von den Betriebsarten abhängigen Teilgrößen zusammen:

$$Y = YP + YI + YD + FEED_FWD$$

Nach der Summation der Anteile findet eine Stellgrößen-Begrenzung statt, so daß gilt:

$$ymin \leq Y \leq ymax$$

Übersicht zur Berechnung der Regler-Anteile

Nachfolgend finden Sie eine Übersicht über die unterschiedliche Berechnung der Regler-Anteile in Abhängigkeit von dem Element trig_err:

Regler-Typ	Regler-Anteile
PI-Regler (ERR ≤ trig_err)	YP und YD für Betriebsart Hand, Halt und Automatik YI für Betriebsart Automatik YI für Betriebsart Hand und Halt
PD-Regler (ERR > trig_err)	YP und YI für Betriebsart Hand, Halt und Automatik YD für Betriebsart Automatik YD für Betriebsart Hand und Halt

PI-Regler: YP und YD für alle Betriebsarten

YP und YD für Betriebsart Hand, Halt, Automatik und Kaskade wird wie folgt ermittelt:

$$YP = gain_i \times ERR$$

$$YD = 0$$

PI-Regler: I-Anteil für Betriebsart Automatik

YI für Betriebsart Automatik wird wie folgt ermittelt (ti > 0):

$$YI = YI_{(old)} + gain_i \times \frac{dt}{ti} \times \frac{ERR + ERR_{(old)}}{2}$$

Der I-Anteil wird nach der Trapezregel gebildet.

PI-Regler: I-Anteil YI für Betriebsart Hand und Halt

YI für Betriebsart Hand und Halt wird wie folgt ermittelt

$$YI = Y - (YP - FEED_FWD)$$

PD-Regler: YP und YI für alle Betriebsarten

YP und YI für Betriebsart Hand, Halt und Automatik wird wie folgt ermittelt

$$YP = gain_d \times ERR$$

$$YI = 0$$

PD-Regler: D-Anteil für Betriebsart Automatik

YD für Betriebsart Automatikwird wie folgt ermittelt:

$$YD = \frac{YD_{(old)} \times td_lag + td \times gain_d \times (ERR - ERR_{(old)})}{dt + dt_lag}$$

PD-Regler: D-Anteil für Betriebsart Hand und Halt

YD für Betriebsart Hand und Halt wird wie folgt ermittelt:

$$YD = 0$$

Laufzeitfehler

Fehlermeldung

Es erfolgt eine Fehlermeldung, wenn

- eine unzulässige Gleitkommazahl am Eingang PV anliegt
- ymax < ymin ist

PDM: Pulsdauermodulation

31

Übersicht

Einleitung

Dieses Kapitel beschreibt den Baustein PDM.

Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Kurzbeschreibung	262
Darstellung	263
Detailbeschreibung	264
Laufzeitfehler	268

Kurzbeschreibung

Verwendung des Bausteins

Die Ansteuerung von Stellgliedern erfolgt nicht nur über analoge Größen sondern auch mittels binärer Stellsignale. Die Umsetzung von analogen Werten in binäre Ausgangssignale erfolgt beispielsweise über Pulsbreitenmodulation (PWM) oder Pulsdauermodulation (PDM).

Dabei soll die eingestellte mittlere Energie des Stellgliedes (Stellenergie) dem analogen Eingangswert (X) des Modulationsbausteins entsprechen.

Funktionsbeschreibung

Der Funktionsbaustein PDM dient zur Umsetzung von analogen Werten in digitale Ausgangssignale.

Bei dem Funktionsbaustein PDM wird ein in seiner Dauer konstantes 1-Signal innerhalb einer vom Analogwert X abhängigen Taktzeit ausgegeben. Die mittlere eingestellte Energie entspricht dem Quotienten aus der Einschaltdauer t_on und der Taktzeitperiode.

Damit die mittlere eingestellte Energie auch der analogen Eingangsgröße X entspricht muß folgendes gelten:

$$T_{period} \sim \frac{1}{X}$$

Als zusätzliche Parameter können EN und ENO projektiert werden.

Allgemeines zur Stellgliedansteuerung

Im allgemeinen wird die binäre Stellgliedansteuerung durch zwei boolsche Signale Y_POS und Y_NEG vorgenommen. Bei einem Motor entspricht der Ausgang Y_POS beispielsweise dem Signal "Rechtslauf" und der Ausgang Y_NEG dem Signal "Linkslauf". Bei einem Ofen können die Ausgänge Y_POS und Y_NEG entsprechend als "Heizen" und "Kühlen" interpretiert werden.

Handelt es sich beim Stellantrieb um einen Motor, so muß u.U. bei nicht selbsthemmenden Getrieben, zur Vermeidung des Nachlaufens, nach dem Einschaltsignal ein Bremsimpuls ausgegeben werden.

Zum Schutz der Leistungselektronik muß nach dem Einschaltimpuls t_on vor dem Bremsimpuls t_brake eine Pausenzeit t_pause eingelegt werden, um Kurzschlüsse zu verhindern.

Formel

Für einen ordnungsgemäßen Betrieb ist bei der Parametrierung folgende Regel zu beachten:

$$t_on + 2 \times t_pause + t_brake \geq \frac{pos_}{neg_} \times t_min$$

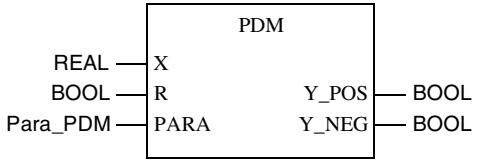
und

$$\frac{pos_}{neg_} \times t_min < \frac{pos_}{neg_} \times t_max$$

Darstellung

Symbol

Darstellung des Bausteins



Parameterbeschreibung PDM

Beschreibung der Bausteinparameter

Parameter	Datentyp	Bedeutung
X	REAL	Eingangsgröße
R	BOOL	Betriebsart Reset
PARA	Para_PDM	Parameter
Y_POS	BOOL	Ausgang für positive X-Werte
Y_NEG	BOOL	Ausgang für negative X-Werte

Parameterbeschreibung Para_PDM

Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
t_on	TIME	Impulsdauer (in s)
t_pause	TIME	Pausenzeit (in s)
t_brake	TIME	Bremszeit (in s)
pos_up_x	REAL	Oberer Eckwert für X positiv
pos_t_min	TIME	Minimale Taktzeit für Y_POS (bei x = pos_up_x) (in s)
pos_lo_x	REAL	Unterer Eckwert für X positiv
pos_t_max	TIME	Maximale Taktzeit für Y_POS (bei x = pos_lo_x) (in s)
neg_up_x	REAL	Oberer Eckwert für X negativ
neg_t_min	TIME	Minimale Taktzeit für Y_NEG (bei x = -neg_up_x) (in s)
neg_lo_x	REAL	Unterer Eckwert für X negativ
neg_t_max	TIME	Maximale Taktzeit für Y_NEG (bei x = -neg_lo_x) (in s)

Detailbeschreibung

Arbeitsweise des Bausteins

Die Impulsdauer t_on legt die Zeitdauer fest, in welcher der Ausgang Y_POS bzw. Y_NEG 1-Signal besitzt. Bei positivem Eingangssignal X wird der Ausgang Y_POS gesetzt, bei negativem Eingangssignal X wird der Ausgang Y_NEG gesetzt. Immer nur ein Ausgang führt 1-Signal. Zwischen Stell- und Bremsimpuls muß zum Schutz der Leitungselektronik (Verhindern des gleichzeitigen Zündens antiparallel geschalteter Stromrichterventile) eventuell eine frei vorgebbare Pausenzeit von t_pause = 10 oder 20 ms eingelegt werden.

An die Impulsdauer schließt sich nach einer Pausenzeit t_pause ein möglicher Bremsimpuls für die Zeitdauer t_brake an. Innerhalb der Pausenzeit führen beide Ausgänge 0-Signal. Während der Bremszeit führt der jeweils zum vorherigen Impuls entgegengesetzte Ausgang 1-Signal. Eine Pausenzeit von t_pause = 20 ms (t_pause = 0.02) entspricht einem Aussetzen der Zündwinkelsteuerung um zwei Halbwellen. Damit ist ein genügend großer Sicherheitsabstand zur Verhinderung von Kurzschlüssen bzw. das Ansprechen der Schutzbeschaltung infolge des Zündens antiparalleler Ventile gewährleistet.

Anschließend folgt eine Zeit in der beide Ausgänge 0-Signal führen (Wartezeit).

Periodendauer

t_{period}

Diese Wartezeit zusammen mit der Impulszeit, der Pausenzeit und der Bremszeit ergeben eine Periodendauer t_{period}, die in Abhängigkeit von lo_x und t_min nach folgenden Gleichungen berechnet wird:

Voraus-setzungen	Gleichung	Erläuterung der Formelgrößen
lo_x <> 0	$t_{period} = t_0 + \frac{K}{X}$	$K = (t_max - t_min) \times \frac{up_x \times lo_x}{up_x - lo_x}$ $t_0 = t_max - \frac{K}{lo_x}$
lo_x = 0 t_min > 0	$t_{period} = \frac{K}{X - X0}$	$X0 = \frac{t_max \times lo_x - t_min \times up_x}{t_max - t_min}$ $K = t_min \times (up_x - X0)$
lo_x = 0 t_min = 0	$t_{period} = t_max \times \left(1 - \frac{X}{up_x}\right)$	

In allen drei Fällen gilt:

Voraussetzung	lo_x	up_x	t_min	t_max
$X \geq \lvert pos_lo_x \rvert$	pos_lo_x	pos_up_x	pos_t_min	pos_t_max
$X \geq -\lvert neg_lo_x \rvert$	neg_lo_x	neg_up_x	neg_t_min	neg_t_max

Hinweis: Von den Parametern up_x (-pos/-neg) und lo_x (-pos/-neg) wird jeweils der Betrag ausgewertet.

Taktzeit

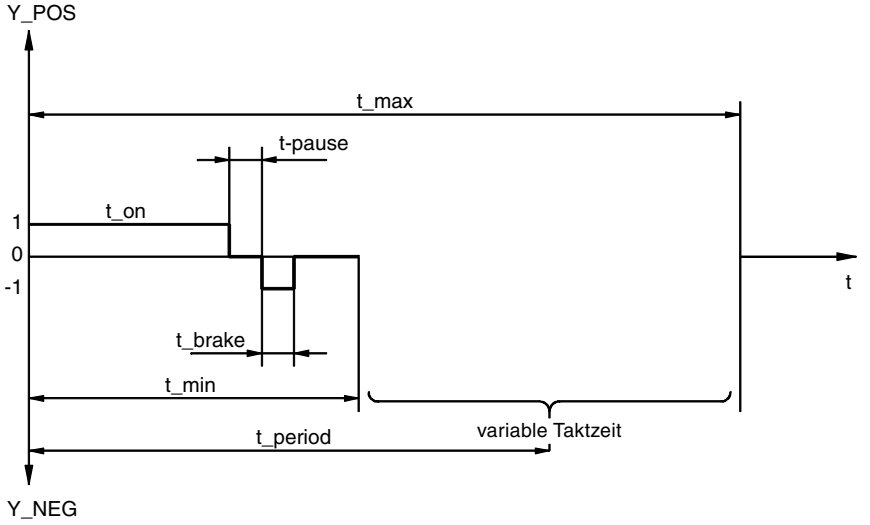
Der Parameter t_min - für jeden Ausgang gibt es einen separaten Wert - gibt die minimale Periodendauer an, d.h. die Zeitdauer, die vom Beginn eines Stellimpulses bis zum Beginn des nächsten Stellimpulses vergeht. Diese Zeitdauer stellt sich ein, wenn der Eingang X den Wert up_x überschreitet - auch hier gibt es für jeden Ausgang einen separaten Wert.

Der Parameter t_max begrenzt die maximale Periodendauer nach oben hin. Unterschreitet der Eingang den Wert pos_lo_x bzw. neg_lo_x, wird solange kein Stellimpuls ausgegeben, bis der Eingang den Wert pos_lo_x bzw neg_lo_x wieder überschreitet. Im Prinzip wird durch die Werte pos_lo_x und neg_lo_x eine Unempfindlichkeitszone festgelegt, in welcher der Funktionsbaustein seine Ausgänge nicht mehr setzt.

Die Parameter (pos_t_min, pos_up_x) und (pos_t_max, pos_lo_x) gelten für positive Eingangssignale X. Hierbei wird der Ausgang Y_POS gesetzt. Die Parameter (neg_t_min, neg_up_x) und (neg_t_max, neg_lo_x) gelten für negative Eingangssignale X. Hierbei wird der Ausgang Y_NEG gesetzt.

Darstellung der Zeitverhältnisse

Ein Verhältnis der Zeiten zueinander zeigt folgendes Bild:

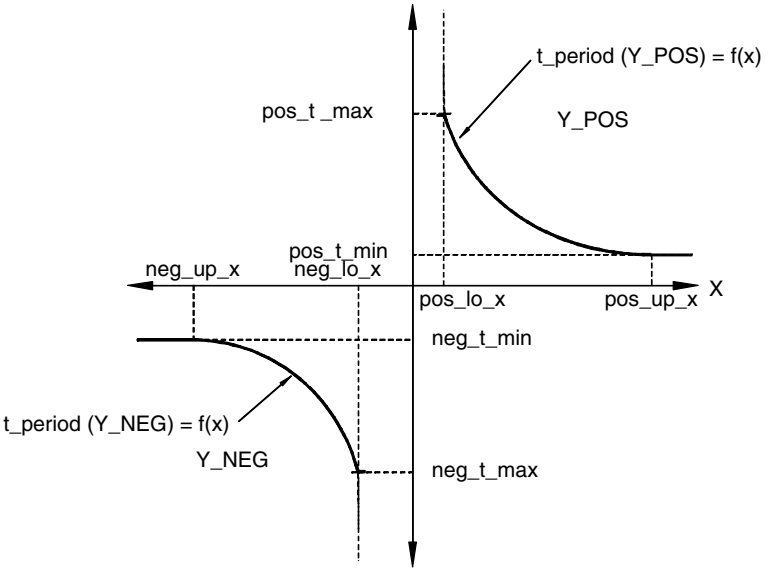


Abhängigkeit der Zeitdauer

Die Abhängigkeit der Zeitdauer, in welcher der Ausgang Y_POS (Y_NEG) 1-Signal hat, von der Eingangsgröße X ist im Bild "*Abhängigkeit der Ausgänge von X, S. 267*" und Bild "*Abhängigkeit der Ausgänge von X (Sonderfall), S. 267*" dargestellt.

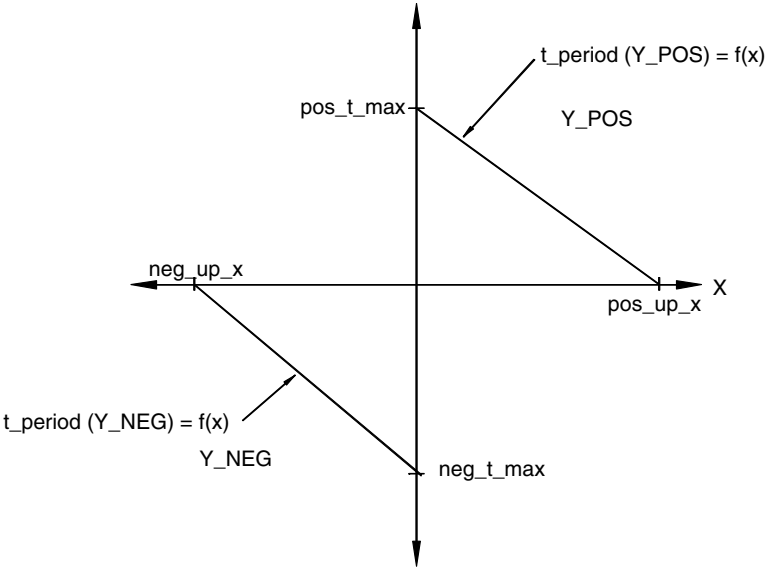
Abhängigkeit der Ausgänge von X

In folgendem Bild ist die Abhängigkeit der Ausgänge von X dargestellt:



Abhängigkeit der Ausgänge von X (Sonderfall)

In nachfolgendem Bild ist der Sonderfall t_min = 0, lo_x = 0 dargestellt:



Betriebsarten

Bei der Betriebsart Reset R = 1 werden die Ausgänge Y_POS und Y_NEG auf Null gesetzt. Die internen Zeitzähler werden ebenfalls normiert, so daß der Funktionsbaustein beim Übergang auf R = 0 mit der Ausgabe eines neuen 1-Signals am zugehörigen Ausgang beginnt.

Randbedingungen

Wird der Funktionsbaustein PDM zusammen mit einem PID-Regler betrieben, so sollte die maximale Periodendauer t_max so gewählt werden, daß sie der Abtastzeit des PID-Reglers entspricht. Dann ist nämlich gewährleistet, daß jedes neue Stellsignal vom PID-Regler innerhalb der Periodendauer vollständig bearbeitet werden kann.

Die Abtastzeit des PDM sollte danach gewählt werden, wie fein die Impulszeit zur Periodendauer steht. Durch sie wird der minimal mögliche Stellimpuls festgelegt.

Es empfiehlt sich minimal folgendes Verhältnis:

t_max/Abtastzeit (PDM) ≥ 10

Laufzeitfehler

Fehlermeldung

Es erfolgt eine Fehlermeldung, wenn

- $|up_x| \leq |lo_x|$
- $t_max \leq t_min$

EFB-Beschreibungen (PI bis Z)

III

Übersicht

Einleitung

Die EFB-Beschreibungen sind in alphabetischer Reihenfolge dokumentiert.

Hinweis: Die Anzahl der Eingänge einiger EFBs kann durch vertikale Größenänderung des FFB-Symbols auf max. 32 erhöht werden. Um welche EFBs es sich dabei handelt entnehmen Sie bitte der Beschreibung der einzelnen EFBs.

Inhalt dieses Teils

Dieser Teil enthält die folgenden Kapitel:

Kapitel	Kapitelname	Seite
32	PI: PI-Regler	271
33	PI1: PI-Regler	281
34	PI_B: Einfacher PI-Regler	291
35	PID: PID-Regler	305
36	PID1: PID-Regler	319
37	PID_P: PID-Regler mit paralleler Struktur	331
38	PID_PF: PID-Regler mit paralleler Struktur	341
39	PIDFF: Kompletter PID-Regler	351
40	PIDP1: PID-Regler mit paralleler Struktur	379
41	PIP: PIP-Kaskaden-Regler	389
42	PPI: PPI-Kaskaden-Regler	401
43	PWM: Pulsbreitenmodulation	411
44	PWM1: Pulsbreitenmodulation	421
45	QDTIME: Totzeitglied	429
46	QPWM: Pulsbreitenmodulation (einfach)	435
47	RAMP: Rampengenerator	443
48	RATIO: Verhältnisregler	449
49	SCALING: Skalierung	455
50	SCON3: Dreipunkt-Schrittregler	459
51	SERVO: Steuerung für elektrische Servomotoren	465
52	SMOOTH_RATE: Differenzierer mit Glättung	481
53	SP_SEL: Sollwertschalter	487
54	SPLRG: Steuerung von 2 Stellgliedern	495
55	STEP2: Zweipunktregler	501
56	STEP3: Dreipunktregler	507
57	SUM_W: Summierer	513
58	THREEPOINT_CON1: Dreipunkt-Regler	515
59	THREE_STEP_CON1: Dreipunkt-Schrittregler	523
60	TOTALIZER: Integrierer	529
61	TWOPOINT_CON1: Zweipunkt-Regler	539
62	VEL_LIM: Anstiegsbegrenzer	545
63	VLIM: Anstiegsbegrenzer 1. Ordnung	551

PI: PI-Regler

32

Übersicht

Einleitung

Diese Kapitel beschreibt den Baustein PI.

Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Kurzbeschreibung	272
Darstellung	273
Formeln	275
Parametrierung	276
Betriebsarten	278
Beispiel zum PI-Regler	279
Laufzeitfehler	280

Kurzbeschreibung

Funktionsbeschreibung

Der Funktionsbaustein stellt einen einfachen PI-Regler dar.

Zwischen Führungsgröße SP und der Regelgröße PV wird eine Regeldifferenz ERR gebildet. Diese Regeldifferenz ERR bewirkt eine Änderung der Stellgröße Y.

Als zusätzliche Parameter können EN und ENO projektiert werden.

Eigenschaften

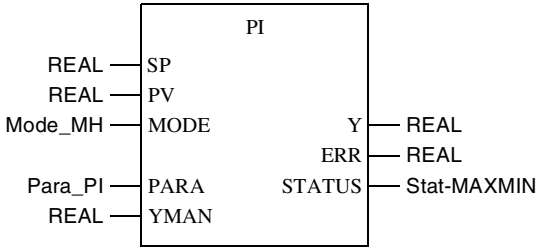
Der Funktionsbaustein besitzt folgende Eigenschaften:

- Betriebsarten Hand, Halt, Automatik
- stoßfreie Umschaltung Hand/Automatik
- Stellgrößen-Begrenzung
- Antiwindup-Reset (bei aktivem I-Anteil)

Darstellung

Symbol

Darstellung des Bausteins:



Parameterbeschreibung PI

Beschreibung der Bausteinparameter

Parameter	Datentyp	Bedeutung
SP	REAL	Sollwerteingang / Führungsgröße
PV	REAL	Istwerteingang / Regelgröße
MODE	Mode_MH	Betriebsart
PARA	Para_PI	Parameter
YMAN	REAL	Handwert
Y	REAL	Stellgröße
ERR	REAL	Regeldifferenz
STATUS	Stat_MAXMIN	Status des Ausgangs Y

Parameterbeschreibung Mode_MH

Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
man	BOOL	"1": Betriebsart Hand
halt	BOOL	"1": Betriebsart Halt

Parameterbeschreibung Para_PI

Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
gain	REAL	Proportionalbeiwert (Verstärkungsfaktor)
ti	TIME	Nachstellzeit
ymax	REAL	Obere Stellgrenze
ymin	REAL	Untere Stellgrenze

Parameterbeschreibung Stat_MAXMIN

Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
qmax	BOOL	"1" = Y hat obere Stellgrenze erreicht
qmin	BOOL	"1" = Y hat untere Stellgrenze erreicht

Formeln

Übertragungsfunktion

Die Übertragunsfunktion lautet:

$$G(s) = gain \times \left(1 + \frac{1}{ti \times s}\right)$$

Berechnungsformeln

Die Berechnungsformeln lauten:

$$YP = gain \times ERR$$

$$YI_{(new)} = YI_{(old)} + gain \times \frac{dt}{ti} \times \frac{ERR_{(new)} + ERR_{(old)}}{2}$$

Ausgangssignal Y

Das Ausgangssignal Y ergibt sich dann zu:

$Y = YP + YI$

Der I-Anteil wird nach der Trapezregel gebildet.

Erläuterung der Formelgrößen

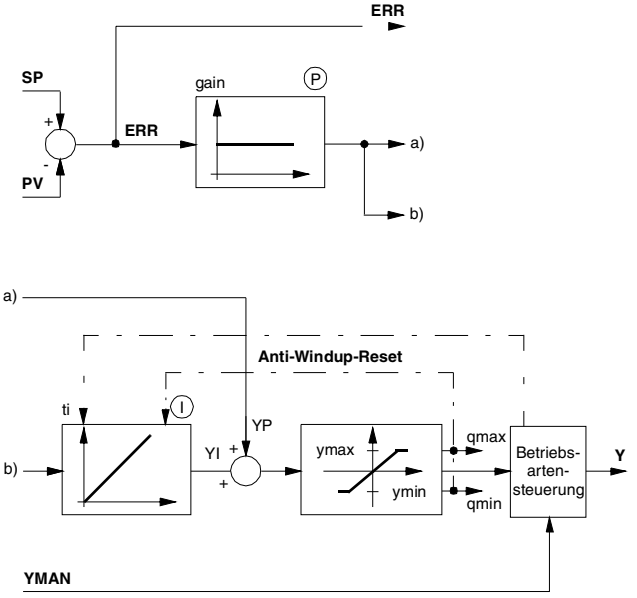
In der folgenden Tabelle finden Sie die Bedeutung der Formelgrößen:

Größe	Bedeutung
dt	Aktuelle Abtastzeit
ERR	Regeldifferenz (SP - PV)
$ERR_{(old)}$	Wert der Regeldifferenz aus dem vorherigen Abtastschritt
YI	I-Anteil
YP	P-Anteil

Parametrierung

Strukturbild

Nachfolgend finden Sie das Strukturbild des PI-Reglers:



Parametrierung

Die Struktur des PI-Reglers ist im obigen *Strukturbild, S. 276* dargestellt. Die Parametrierung des Funktionsbausteins erfolgt zunächst durch die reinen PI-Parameter: den Proportionalbeiwert gain und der Nachstellzeit ti.

Der I-Anteil kann durch Nullsetzen von ti abgeschaltet werden.

Die Werte ymax und ymin begrenzen den Ausgang nach oben sowie nach unten. Damit gilt ymin ≤ Y ≤ ymax.

Das Erreichen der Grenzwerte, sowie eine Begrenzung des Ausgangssignals wird durch die beiden Merker qmax und qmin angezeigt:

- qmax = 1 wenn Y ≥ ymax
- qmin = 1 wenn Y ≤ ymin

Stellgrößen-Begrenzung

Nach der Summation der Anteile findet eine Stellgrößen-Begrenzung statt, so daß gilt: ymin ≤ Y ≤ ymax

Antiwindup-Reset

Findet eine Stellgrößen-Begrenzung statt, soll durch Antiwindup-Reset dafür gesorgt werden, daß der integrale Anteil "nicht über alle Grenzen wachsen kann". Die Anti-Windup-Maßnahme wird nur dann durchgeführt, wenn der I-Anteil des Reglers nicht abgeschaltet ist. Die Grenzen für Anti-Windup sind hier dieselben, wie für die Stellgrößenbegrenzung. Die Anti-Windup-Reset-Maßnahme korrigiert den I-Anteil in der Form, daß gilt: $ymin - YP \leq YI \leq ymax - YP$

Betriebsarten

Auswahl der Betriebsarten

Es gibt drei Betriebsarten, die über die Elemente man und halt ausgewählt werden.

Betriebsart	man	halt
Automatik	0	0
Hand	1	1 oder 0
Halt	0	1

Betriebsart Automatik

In der Betriebsart Automatik wird der Stellausgang Y durch die Regelung bestimmt, in Abhängigkeit von Regelgröße PV und Führungsgröße SP. Die Stellgröße wird durch ymax und ymin begrenzt. Die Stellgrenzen sind gleichzeitig auch Grenzen für den Anti-Windup-Reset.

Die Umschaltung von Automatik nach Hand ist normalerweise nicht stoßfrei, da der Ausgang Y jeden Wert zwischen ymax und ymin haben kann und Y bei der Umschaltung direkt den Wert YMAN annimmt.

Soll die Umschaltung von Automatik nach Hand trotzdem stoßfrei erfolgen gibt es zwei Möglichkeiten, die für einen PID-Regler (siehe *Umschaltung von Automatik nach Hand, S. 312*) beispielhaft erklärt sind.

Betriebsart Hand

In der Betriebsart Hand wird der Handstellwert YMAN fest auf den Stellausgang Y übertragen. Die Stellgröße wird jedoch durch ymax und ymin begrenzt. Die internen Größen werden so nachgeführt, daß der Regler (bei zugeschaltetem I-Anteil) stoßfrei von Hand auf Automatik umgeschaltet werden kann. Die Stellgrenzen sind gleichzeitig auch Grenzen für den Anti-Windup-Reset

Betriebsart Halt

In der Betriebsart Halt bleibt der Reglerausgang, wie er vorgefunden wird, der Funktionsbaustein verändert den Reglerausgang Y nicht (Regler bleibt stehen), d.h. Y = Y(old). Die internen Größen werden so nachgeführt, daß die Summe ihrer Anteile der Stellgröße entsprechen, so daß der Regler ohne Stoß von seiner aktuellen Lage aus weiterfährt. Die Stellgrenzen sind gleichzeitig auch Grenzen für den Anti-Windup-Reset. Die Betriebsart Halt eignet sich unter anderem auch dafür, die Stellgröße Y über ein externes Bediengerät zu stellen, wobei die internen Anteile im Regler richtig nachgeführt werden.

Beispiel zum PI-Regler

Beispiel

In nachfolgendem Bild (siehe *Sprungantwort des PI-Reglers, S. 279*) ist als Beispiel die Sprungantwort des PI-Reglers dargestellt.

Im ersten Teil des Bildes ist die Reaktion des Funktionsbausteins auf die Betriebsart MAN dargestellt: Der Ausgang Y springt auf den Wert YMAN.

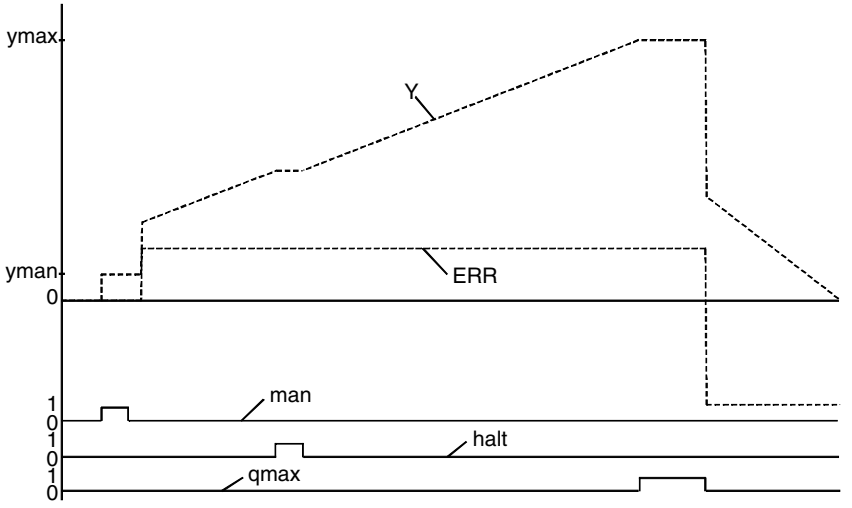
Im zweiten Teil des Bildes ist die Reaktion des Funktionsbausteins im Automatik-Betrieb (MAN = 0 und HALT= 0) sowohl mit positiver Regeldifferenz ERR, als auch mit negativer Regeldifferenz ERR dargestellt. Bei konstanter positiver Regeldifferenz steigt der Y rampenförmig an, bis die obere Stellgrenze erreicht wird.

Y wird im folgenden auf den Wert ymax begrenzt. Die Begrenzung wird durch qmax gemeldet. Die Regeldifferenz springt nun auf einen negativen Wert, der betragsmäßig größer ist, als der vorherige Wert.

Durch den P-Anteil springt der Ausgang um den Wert $\mathrm{gain} \times (\mathrm{ERR}_{(\mathrm{new})} - \mathrm{ERR}_{(\mathrm{old})})$; danach fällt Y rampenförmig ab. Die Steigung ist hier betragsmäßig größer als vorher bei positiver Regeldifferenz. Dies ist auf den jetzt betragsmäßig größeren Wert der Regeldifferenz zurückzuführen.

Sprungantwort des PI-Reglers

Darstellung der Sprungantwort des PI-Reglers



Laufzeitfehler

Fehlermeldung

Es erfolgt eine Fehlermeldung, wenn

- eine unzulässige Gleitkommazahl am Eingang YMAN oder X anliegt,
- ymax < ymin ist.

PI1: PI-Regler

33

Übersicht

Einleitung

Diese Kapitel beschreibt den Baustein PI1.

Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Kurzbeschreibung	282
Darstellung	283
Formeln	284
Parametrierung	285
Betriebsarten	287
Beispiel zum PI1-Regler	288
Laufzeitfehler	289

Kurzbeschreibung

Funktionsbeschreibung

Der Funktionsbaustein stellt einen einfachen PI-Regler dar.

Zwischen Sollwert SP und der Regelgröße PV wird eine Regeldifferenz ERR gebildet. Diese Regeldifferenz ERR bewirkt eine Änderung der Stellgröße Y.

Als zusätzliche Parameter können EN und ENO projektiert werden.

Eigenschaften

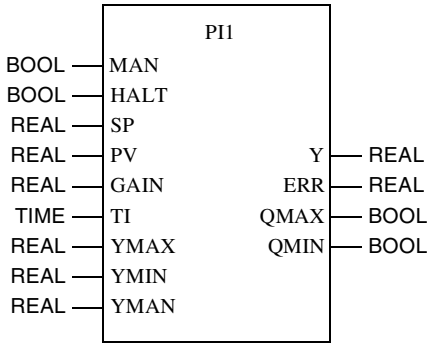
Der Funktionsbaustein besitzt folgende Eigenschaften:

- Betriebsarten Hand, Halt, Automtik
- stoßfreie Umschaltung Hand/Automatik
- Stellgrößen-Begrenzung
- Anti-Windup-Reset
- Anti-Windup-Maßnahme nur bei aktivem I-Anteil

Darstellung

Symbol

Darstellung des Bausteins:



Parameterbeschreibung

Beschreibung der Bausteinparameter

Parameter	Datentyp	Bedeutung
MAN	BOOL	"1": Betriebsart Hand
HALT	BOOL	"1": Betriebsart Halt
SP	REAL	Sollwerteingang
PV	REAL	Eingangsgröße
GAIN	REAL	Proportionalbeiwert (Verstärkungsfaktor)
TI	TIME	Nachstellzeit
YMAX	REAL	Obere Stellgrenze
YMIN	REAL	Untere Stellgrenze
YMAN	REAL	Handwert
Y	REAL	Stellgröße
ERR	REAL	Ausgang Regeldifferenz
QMAX	BOOL	"1" = Ausgang Y hat obere Stellgrenze erreicht
QMIN	BOOL	"1" = Ausgang Y hat untere Stellgrenze erreicht

Formeln

Übertragungs-funktion

Die Übertragunsfunktion lautet:

$$G(s) = GAIN \times \left(1 + \frac{1}{TI \times s}\right)$$

Der I-Anteil kann durch Nullsetzen von TI abgeschaltet werden.

Berechnungs-formeln

Die Berechnungsformeln lauten:

$$YP = GAIN \times ERR$$

$$YI_{(new)} = YI_{(old)} + GAIN \times \frac{dt}{TI} \times \frac{ERR_{(new)} + ERR_{(old)}}{2}$$

Ausgangssignal Y

Das Ausgangssignal Y ergibt sich dann zu:

$$Y = YP + YI$$

Der I-Anteil wird nach der Trapezregel gebildet.

Erläuterung der Formelgrößen

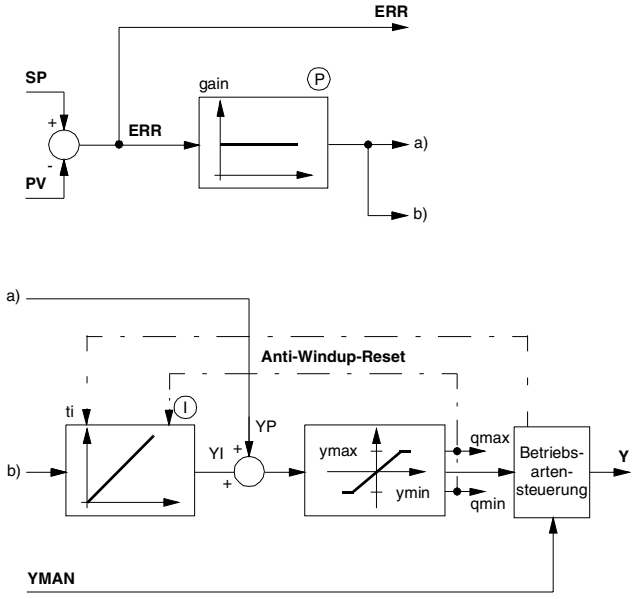
In der folgenden Tabelle finden Sie die Bedeutung der Formelgrößen:

Größe	Bedeutung
dt	Aktuelle Abtastzeit
ERR	Regeldifferenz (SP - PV)
$ERR_{(old)}$	Wert der Regeldifferenz aus dem vorherigen Abtastschritt
YI	I-Anteil
YP	P-Anteil

Parametrierung

Strukturbild

Nachfolgend finden Sie das Strukturbild des PI1-Reglers:



Parametrierung

Die Struktur des PI1-Reglers ist im obigen *Strukturbild, S. 285* dargestellt. Die Parametrierung des Funktionsbausteins erfolgt zunächst durch die reinen PI-Parameter: den Proportionalbeiwert GAIN und der Nachstellzeit TI.

Die Grenzen YMAX und YMIN begrenzen den Ausgang nach oben sowie nach unten. Damit gilt YMIN ≤ Y ≤ YMAX.

Das Erreichen der Grenzwerte, bzw. eine Begrenzung des Ausgangssignals wird durch die beiden Merker QMAX und QMIN angezeigt:

- QMAX = 1 wenn Y ≥ YMAX
- QMIN = 1 wenn Y ≤ YMIN

Stellgrößen-Begrenzung

Nach der Summation der Anteile findet eine Stellgrößen-Begrenzung statt, so daß gilt: YMIN ≤ Y ≤ YMAX

Antiwindup-Reset

Findet eine Stellgrößen-Begrenzung statt, soll durch Antiwindup-Reset dafür gesorgt werden, daß der integrale Anteil "nicht über alle Grenzen wachsen kann". Die Anti-Windup-Maßnahme wird nur dann durchgeführt, wenn der I-Anteil des Reglers nicht abgeschaltet ist. Die Grenzen für Anti-Windup sind hier dieselben, wie für die Stellgrößenbegrenzung. Die Anti-Windup-Reset-Maßnahme korrigiert den I-Anteil in der Form, daß gilt: YMIN - YP ≤ YI ≤ YMAX - YP

Betriebsarten

Auswahl der Betriebsarten

Es gibt drei Betriebsarten, die über die Eingänge MAN und HALT ausgewählt werden.

Betriebsart	MAN	HALT
Automatik	0	0
Hand	1	1 oder 0
Halt	0	1

Betriebsart Automatik

In der Betriebsart Automatik wird der Stellausgang Y durch die Regelung bestimmt, in Abhängigkeit von Regelgröße PV und Führungsgröße SP. Die Stellgröße wird durch YMAX und YMIN begrenzt. Die Stellgrenzen sind gleichzeitig auch Grenzen für den Anti-Windup-Reset.

Die Umschaltung von Automatik nach Hand ist normalerweise nicht stoßfrei, da der Ausgang Y jeden Wert zwischen YMAX und YMIN haben kann und Y bei der Umschaltung direkt den Wert YMAN annimmt.

Soll die Umschaltung von Automatik nach Hand trotzdem stoßfrei erfolgen gibt es zwei Möglichkeiten, die für einen PID1-Regler (siehe *Umschaltung von Automatik nach Hand, S. 326*) beispielhaft erklärt sind.

Betriebsart Hand

In der Betriebsart Hand wird der Handstellwert YMAN fest auf den Stellausgang Y übertragen. Die Stellgröße wird jedoch durch YMAX und YMIN begrenzt. Die internen Größen werden so nachgeführt, daß der Regler (bei zugeschaltetem I-Anteil) stoßfrei von Hand auf Automatik umgeschaltet werden kann. Die Stellgrenzen sind gleichzeitig auch Grenzen für den Anti-Windup-Reset

Betriebsart Halt

In der Betriebsart Halt bleibt der Reglerausgang, wie er vorgefunden wird, der Funktionsbaustein verändert den Reglerausgang Y nicht (Regler bleibt stehen), d.h. Y = Y(old). Die internen Größen werden so nachgeführt, daß die Summe ihrer Anteile der Stellgröße entsprechen, so daß der Regler ohne Stoß von seiner aktuellen Lage aus weiterfährt. Die Stellgrenzen sind gleichzeitig auch Grenzen für den Anti-Windup-Reset.

Beispiel zum PI1-Regler

Beispiel

In nachfolgendem Bild (siehe *Sprungantwort des PI1-Reglers, S. 288*) ist als Beispiel die Sprungantwort des PI1-Reglers dargestellt.

Im ersten Teil des Bildes ist die Reaktion des Funktionsbausteins auf die Betriebsart MAN dargestellt: Der Ausgang Y springt auf den Wert YMAN.

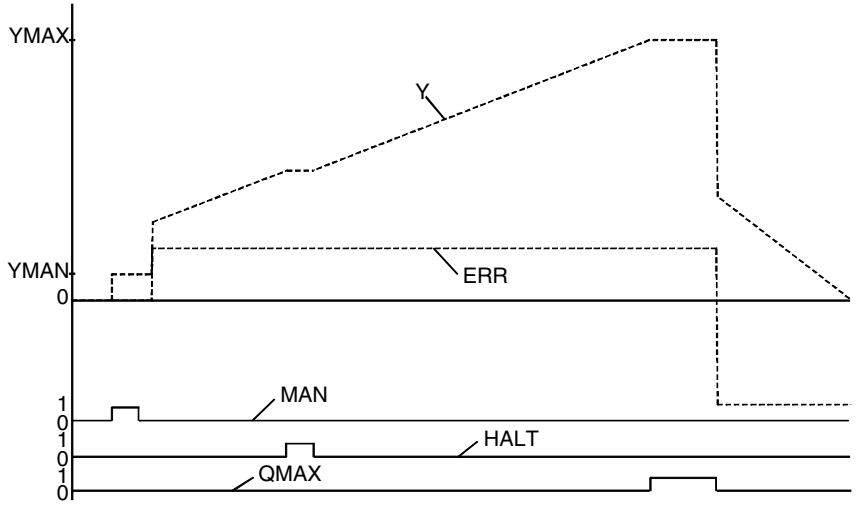
Im zweiten Teil des Bildes ist die Reaktion des Funktionsbausteins im Automatik-Betrieb (MAN = 0 und HALT= 0) sowohl mit positiver Regeldifferenz ERR, als auch mit negativer Regeldifferenz ERR dargestellt. Bei konstanter positiver Regeldifferenz steigt der Y rampenförmig an, bis die obere Stellgrenze erreicht wird.

Der Ausgang wird im folgenden auf den Wert YMAX begrenzt. Die Begrenzung wird im Merker QMAX gemeldet. Die Regeldifferenz springt nun auf einen negativen Wert, der betragsmäßig größer ist, als der vorherige Wert.

Durch den P-Anteil springt der Ausgang um den Wert $GAIN \times (ERR_{(new)} - ERR_{(old)})$; danach fällt Y rampenförmig ab. Die Steigung ist hier betragsmäßig größer als vorher bei positiver Regeldifferenz. Dies ist auf den jetzt betragsmäßig größeren Wert der Regeldifferenz zurückzuführen.

Sprungantwort des PI1-Reglers

Darstellung der Sprungantwort des PI1-Reglers



Laufzeitfehler

Fehlermeldung Bei YMAX < YMIN erfolgt eine Fehlermeldung.

PI_B: Einfacher PI-Regler

34

Übersicht

Einleitung

Diese Kapitel beschreibt den Baustein PI_B.

Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Kurzbeschreibung	292
Darstellung	293
Formeln	295
Parametrierung	296
Detaillierte Gleichungen	300
Laufzeitfehler	302

Kurzbeschreibung

Funktionsbeschreibung

Der Funktionsbaustein PI_B stellt einen PI-Algorithmus mit gemischter Struktur (seriell/parallel) dar. Seine Funktionalitäten sind aus denen des Funktionsbausteins PIDFF (siehe *PIDFF: Kompletter PID-Regler, S. 351*) hervorgegangen. Diese Funktionalitäten ermöglichen es dem Funktionsbaustein, die meisten klassischen Regelapplikationen durchzuführen, ohne an Benutzerfreundlichkeit einzubüßen oder zu viele Systemressourcen in Anspruch zu nehmen. Bei schwierigen Regelaufgaben, die erweiterte Regelfunktionen benötigen, sollte jedoch der Baustein PIDFF verwendet werden.

Als zusätzliche Parameter können EN und ENO projektiert werden.

Funktionalitäten

Die wichtigsten Funktionalitäten des Funktionsbausteins PI_B sind folgende:

- Berechnung des Proportional- und des Integralanteils in inkrementaler Form
- Istwert, Sollwert und Ausgangswert in physikalischen Einheiten
- direkte oder inverse Aktion
- Möglichkeit der Aufschaltung eines bausteinexternen I-Anteils (RCPY-Eingang)
- Unempfindlichkeitszone auf Abweichung
- Inkrementalwert- und Absolutwertausgang
- Oberer und unterer Grenzwert des Ausgangssignals
- Ausgangs-Offset
- Auswahl der Betriebsart Hand/Automatik
- Tracking-Betrieb
- Oberer und unterer Grenzwert des Sollwertes

Erweiterte Funktionalitäten

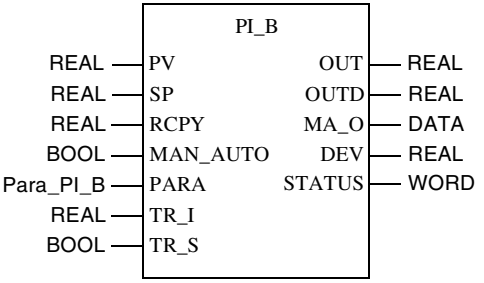
Wie beim PIDFF können durch den zusätzlichen Einsatz anderer Funktionsbausteine diese Funktionalitäten erweitert werden:

- Automatische Reglereinstellung duch den Baustein AUTOTUNE
- Auswahl eines internen oder externen Sollwertes durch den Baustein SP_SEL
- Steuerung des Handbetriebes der abgetasteten Regelkreise (siehe *Abtasten, S. 36*) mit dem Funktionsbaustein MS

Darstellung

Symbol

Darstellung des Bausteins:



Parameterbeschreibung PI_B

Beschreibung der Bausteinparameter

Parameter	Datentyp	Bedeutung
PV	REAL	Istwert (Process Value)
SP	REAL	Sollwert (Set Point)
RCPY	REAL	Kopie der effektiven Stellgröße
MAN_AUTO	BOOL	Regler-Betriebsart: "1" : Automatik-Betrieb "0" : Handbetrieb
PARA	Para_PI_B	Parameter
TR_I	REAL	Initialisierungseingang
TR_S	BOOL	Initialisierungsbefehl
OUT	REAL	Stellgliedausgang
OUTD	REAL	Differentialausgang: Unterschied zwischen Ausgang der aktuellen und der vorherigen Ausführung
MA_O	BOOL	Aktuelle Betriebsart des Funktionsbausteins: "1" : Betriebsart Automatik "0" : andere Betriebsart (d.h. Handbetrieb oder Tracking)
DEV	REAL	Wert der Abweichung (PV - SP)
STATUS	WORD	Statuswort

Parameterbeschreibung Para_PI_B

Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
id	UINT	Reserviert für automatische Reglereinstellung
pv_inf	REAL	Untere Grenze des Istwertbereichs
pv_sup	REAL	Obere Grenze des Istwertbereichs
out_inf	REAL	Untere Grenze des Ausgangswertbereichs
out_sup	REAL	Obere Grenze des Ausgangswertbereichs
rev_dir	BOOL	"1" : direkte Aktion des PID-Reglers "0" : inverse Aktion des PID-Reglers
en_rcpy	BOOL	"1" : der RCPY-Eingang wird verwendet
kp	REAL	Proportionalbeiwert (Verstärkungsfaktor)
ti	TIME	Nachstellzeit
dband	REAL	Unempfindlichkeitszone auf Abweichung
outbias	REAL	Manueller Ausgleich der statischen Abweichung

Formeln

Übertragungsfunktion

Die Übertragungsfunktion lautet:

$$OUT = kp \times \left(1 + \frac{1}{ti \times p}\right) \times IN$$

Berechnungsformeln

Die tatsächlich verwendeten Formeln variieren, je nachdem, ob der Funktionsbaustein den inkrementalen oder den absoluten Algorithmus verwendet.

Sehr vereinfacht kann der Funktionsbaustein eine der folgenden Formeln verwenden:

Algorithmus	ti	Formen
Absolut	0	$OUT = TermP + outbias$ $OUTD = OUT(new) - OUT(old)$
Inkremental	>0	$OUTD = TermP + TermI$ $OUT = OUT(old) + OUTD(new)$

Erläuterung der Formelgrößen

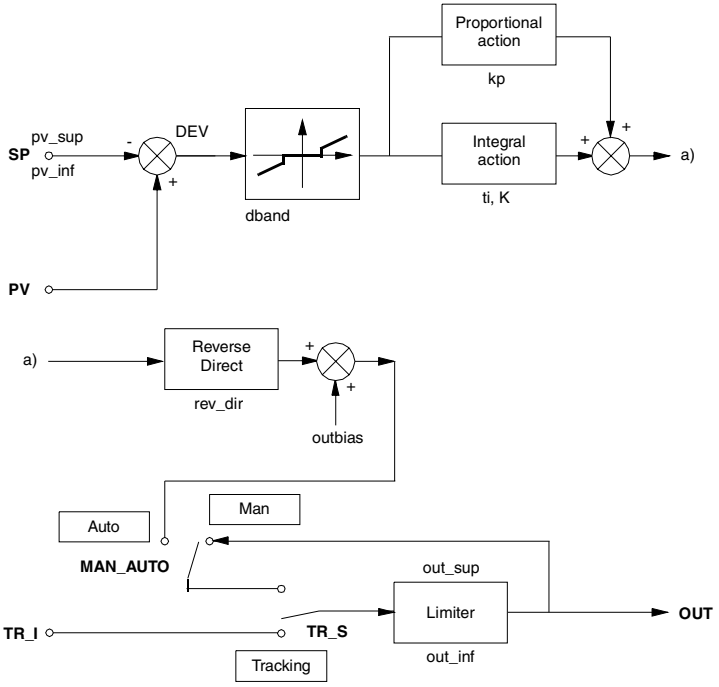
In der folgenden Tabelle finden Sie die Bedeutung der Formelgrößen:

Größe	Bedeutung
(new)	Wert, der bei der aktuellen Ausführung des Funktionsbausteins berechnet wird
(old)	Wert, der bei der vorherigen Ausführung des Funktionsbausteins berechnet wurde
OUT	Absolutwertausgang
OUTD	Inkrementalwertausgang
TermI	Wert des Integralanteils (abhängig vom Algorithmus)
TermP	Wert des Proportionalanteils (abhängig vom Algorithmus)

Parametrierung

Strukturbild des PI_B-Reglers

Strukturbild des PI_B-Reglers



Absolute Algorithmen

Der absolute Algorithmus wird verwendet, wenn kein I-Anteil vorhanden ist (wenn ti = 0). In diesem Fall wird zuerst der Ausgang OUT berechnet und die Ausgangsänderung daraus gefolgert.

Inkrementale Algorithmen

Inkrementale Algorithmen werden verwendet, wenn ein I-Anteil vorhanden ist (d.h. wenn ti > 0). Die Besonderheit dieses Algorithmus besteht darin, zuerst die Ausgangsänderung OUTD zu berechnen und daraufhin den Absolutwertausgang mittels folgender Formel zu bestimmen:

$$OUT(new) = OUT(old) + OUTD$$

Bei diesem Algorithmus kann dem Regler ein SERVO-Funktionsbaustein nachgeschaltet werden, was eine astatische Regelung ermöglicht.

Der inkrementale Algorithmus ermöglicht darüber hinaus die Projektierung eines bausteinexternen Integralanteils bei Regelapplikationen, wo das tatsächlich aufgeschaltete Verhalten vom durch den Regler berechneten Verhalten abweicht (bei offenem Regelkreis). In diesem Fall ist es vorteilhaft, bei der Berechnung den realen Wert zu verwenden. Wenn dieser zur Verfügung steht, ist er dem RCPY-Eingang aufzuschalten und der Parameter en_rcpy auf 1 zu setzen. Bei der Berechnung wird demnach die Gleichung

$$OUT(new) = OUT(old) + OUTD$$

zu

$$OUT(new) = RCPY + OUTD$$

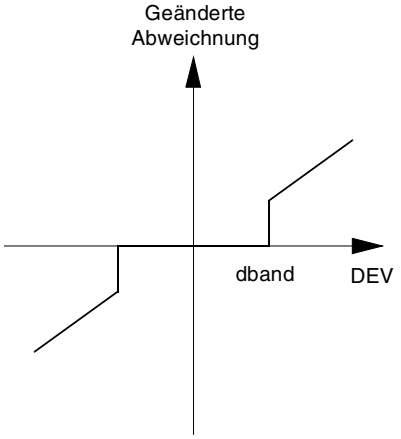
Diese Einrichtung ist insbesondere bei Kaskaden- oder kaskadenähnlicher Regelung vorteilhaft.

Hinweis: Bei Aufschaltung eines externen Integralanteils (en_rcpy=1) ist der Ausgang OUT nicht begrenzt.

Unempfindlichkeitszone auf Abweichung (dband)

Die Unempfindlichkeitszone ermöglicht es bei Erreichen des Arbeitspunktes, kleinere Angleichstöße in bezug auf den Wert des Stellgliedes zu begrenzen: solange die Abweichung unterhalb von dband liegt (in Absolutwerten), legt der Funktionsbaustein bei der Berechnung den Wert Null zugrunde.

Darstellung der Unempfindlichkeitszone auf Abweichung (dband)



Weitere Eigenschaften

Der Baustein besitzt folgende weitere Eigenschaften:

- Die Verwendung des Parameters outbias ermöglicht eine präzise Einstellung am Arbeitspunkt, wenn kein Integralanteil vorhanden ist (ti = 0).
- In allen Betriebsarten ist der Ausgang OUT begrenzt auf den Bereich zwischen out_inf und out_sup. Wenn der vom Funktionsbaustein berechnete Wert (oder vom Anwender im Handbetrieb geschriebene Wert) eine dieser Grenzen übersteigt, wird der Wert von OUT gekappt. Der Inkrementalausgang OUTD berücksichtigt diese Kappung jedoch nie. Dies ermöglicht es dem PI_B, einen SERVO-Funktionsbaustein zu steuern, ohne die Stellgliedposition rückführen zu müssen (kontinuierliche Regelung).
- Die Auswahl direkte/inversre Aktion (Parameter rev_dir) ermöglicht es, die Reglerrichtung der Verbindung Stellglied/Meßprozeß anzupassen.
- Begrenzung des Sollwertes zwischen pv_inf und pv_sup.
- Der Funktionsbaustein kann im reinen Integralbetrieb arbeiten (mit kp = 0).

Betriebsarten

Der Funktionsbaustein PI_B hat 3 Betriebsarten: Automatik, Hand und Tracking. Der Tracking-Betrieb hat Vorrang vor den anderen Betriebsarten.

Diese Betriebsarten können über die Eingänge MAN_AUTO und TR_S ausgewählt werden.

Betriebsart	TR_S	MAN_AUTO	Bedeutung
Automatik	0	1	Die Ausgänge OUT und OUTD entsprechen dem Ergebnis der vom Funktionsbaustein durchgeführten Berechnungen.
Hand	0	0	Der Ausgang OUT wird vom Funktionsbaustein nicht gestellt, so daß der Anwender den Wert direkt ändern kann.
Tracking	1	0 oder 1	Der Eingang TR_I wird auf den Ausgang OUT geführt.

Umschaltung der Betriebsarten

Die Umschaltung Hand → Automatik oder Tracking → Automatik wird wie folgt durchgeführt:

- Beim inkrementalen Algorithmus (ti > 0) erfolgt die Umschaltung stoßfrei.
- Beim absoluten Algorithmus (ti =0) erfolgt die Umschaltung mit Stoß.

Detaillierte Gleichungen

Konvention

Die nachfolgenden Gleichungen verwenden verschiedene Variablen und Funktionen. Die den Bausteinparametern entsprechenden Variablen werden an dieser Stelle nicht erneut beschrieben.

Die wichtigsten Zwischenvariablen und die verwendeten Funktionen werden jedoch in der folgenden Tabelle beschrieben :

Zwischenvariable / Funktion	**Bedeutung**
dt	Zeitintervall seit der letzten Ausführung des Funktionsbausteins
(new)	Wert, der bei der aktuellen Ausführung des Funktionsbausteins berechnet wird
(old)	Wert, der bei der vorherigen Ausführung des Funktionsbausteins berechnet wurde
TermI	Wert des Integralanteils (abhängig vom Algorithmus)
TermP	Wert des Proportionalanteils (abhängig vom Algorithmus)
sense	Regelsinn mit folgenden Wirkrichtungen: • +1 Es handelt sich um eine direkte Aktion (rev_dir = 1), d.h. daß eine positive Abweichung (PV - SP) einen größeren Ausgangswert generiert • -1 Es handelt sich um eine inversre Aktion (rev_dir = 0), d.h. daß eine positive Abweichung (PV - SP) einen kleineren Ausgangswert generiert.
Funktion Δ	$\Delta(x(t)) = x(t) - x(t-1)$
Funktion 'Begrenzen'	Begrenzungsfunktion des Bausteinausgangs

Absoluter Algorithmus

Folgende Gleichungen gelten mit Proportionalreglern (ti = 0);

$$OUT = TermP + outbias$$

$$OUTD = OUT(new) - OUT(old)$$

$$OUT = limiter(OUT)$$

$$TermP = sense \times kp \times DEV$$

Inkrementaler Algorithmus

Folgende Gleichungen gelten mit Reglern des Typs PI (ti > 0);

$$OUTD = TermP + TermI$$

$$OUT = limiter(OUT)$$

Wenn en_rcpy = 0 ist, gilt:

$$OUT = OUT(old) + OUTD(new)$$

Wenn en_rcpy = 1 ist, gilt:

$$OUT = RCPY + OUTD(new)$$

Wert des Proportionalanteils TermP

$$TermP = sense \times kp \times [\Delta(DEV)]$$

Wert des Integralanteils TermI, wenn kp > 0:

$$TermI = sense \times kp \times \frac{dt}{ti} \times DEV$$

Wert des Integralanteils TermI, wenn kp = 0 (reiner Integralbetrieb):

$$TermI = sense \times \frac{out_sup - out_inf}{pv_sup - pv_inf} \times \frac{dt}{ti} \times DEV$$

Laufzeitfehler

Statuswort

Im Statuswort werden folgenden Meldungen angezeigt:

Bit	Bedeutung
Bit 0 = 1	Fehler in einer Berechnung mit Gleitkommawerten
Bit 1 = 1	Erfassung eines nicht zulässigen Wertes an einem der Gleitkommawerteingänge
Bit 2 = 1	Division durch Null bei einer Berechnung mit Gleitkommawerten
Bit 3 = 1	Kapazitätsüberlauf bei einer Berechnung mit Gleitkommawerten
Bit 4 = 1	Folgende Verhalten werden angezeigt: ● Der Eingang SP liegt außerhalb des Bereichs [pv_inf, pv_sup] : zur Berechnung verwendet der Funktionsbaustein den Wert pv_inf bzw. pv_sup. ● Der Parameter kp oder dband ist negativ: der Funktionsbaustein verwendet den Wert 0 anstelle des fehlerhaften Parameterwertes. ● Der outbias-Parameter liegt außerhalb des Bereichs [(out_inf - out_sup), (out_sup - out_inf)]. Der Funktionsbaustein verwendet zur Berechnung den Wert (out_inf - out_sup) bzw. (out_sup - out_inf).
Bit 5 = 1	Der Ausgang OUT hat den unteren Grenzwert out_min erreicht (siehe Hinweis)
Bit 6 = 1	Der Ausgang OUT hat den oberen Grenzwert out_max erreicht (siehe Hinweis)
Bit 7 = 1	Die Grenzwerte pv_inf und pv_sup sind identisch

Hinweis zum Ausgang OUT

Hinweis: Im Handbetrieb bleiben diese Bits nur einen Programmzyklus lang auf 1. Wenn der Anwender für OUT einen Wert eingibt, der eine dieser Grenzen überschreitet, setzt der Funktionsbaustein das Bit 5 oder 6 auf 1 und kappt den vom Bediener eingegebenen Wert. Bei der folgenden Ausführung des Funktionsbausteins liegt der Wert von OUT nicht mehr außerhalb des Bereichs und die Bits 5 und 6 werden wieder auf Null gesetzt.

Fehlermeldung

Ein Fehler wird angezeigt, wenn an einem Eingang ein nicht gleitender Wert erfaßt worden ist, wenn in einer Berechnung mit Gleitkommawerten ein Problem auftritt oder wenn die Grenzwerte pv_inf und pv_sup identisch sind. Die Ausgänge OUT, OUTD, MA_O und DEV bleiben unverändert.

Warnung

In folgenden Fällen wird eine Warnung ausgegeben:

- Einer der Parameter kp oder dband ist negativ: der Funktionsbaustein verwendet den Wert 0 anstelle des Wertes des fehlerhaften Parameters.
- Der Parameter outbias liegt nicht im Bereich [(out_inf - out_sup), (out_sup - out_inf)]. Der Funktionsbaustein verwendet zur Berechnung den Wert (out_inf - out_sup) bzw. (out_sup - out_inf).

PID: PID-Regler

35

Übersicht

Einleitung

Diese Kapitel beschreibt den Baustein PID.

Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Kurzbeschreibung	306
Darstellung	307
Strukturbild des Funktionsbausteins PID	309
Parametrierung des PID-Reglers	310
Betriebsarten	312
Detaillierte Formeln	315
Laufzeitfehler	317

Kurzbeschreibung

Funktionsbeschreibung

Der Funktionsbaustein realisiert einen PID-Regler.

Aufgrund der Führungsgröße SP und der Regelgröße PV wird eine Regeldifferenz ERR gebildet. Diese Regeldifferenz ERR bewirkt eine Änderung der Stellgröße Y.

Als zusätzliche Parameter können EN und ENO projektiert werden.

Eigenschaften

Der Funktionsbaustein besitzt folgenden Eigenschaften:

- realer PID-Regler mit unabhängiger gain, ti, td-Einstellung
- Betriebsarten Hand, Halt, Automatik
- stoßfreie Umschaltung Hand/Automatik
- Stellgrößen-Begrenzung im Automatik-Betrieb
- Einzeln zuschaltbarer P-, I- und D-Anteil
- Anti-Windup-Reset
- Anti-Windup-Maßnahme nur bei aktivem I-Anteil
- definierbare Verzögerung des D-Anteils
- D-Anteil umschaltbar auf Regelgröße PV oder Regeldifferenz ERR

Übertragungsfunktion

Die Übertragungsfunktion lautet:

$$G(s) = gain \times \left(1 + \frac{1}{ti \times s} + \frac{td \times s}{1 + td_lag \times s}\right)$$

YD
YI
YP

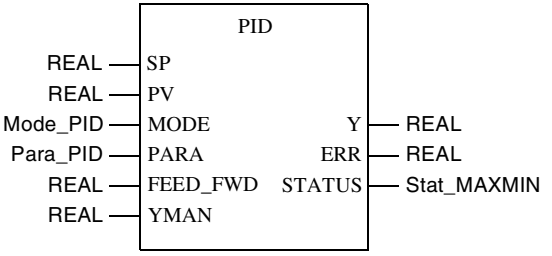
Erläuterung der Größen:

Größe	Bedeutung
YD	D-Anteil (nur bei en_d = 1)
YI	I-Anteil (nur bei en_i = 1)
YP	P-Anteil (nur bei en_p = 1)

Darstellung

Symbol

Darstellung des Bausteins:



Parameterbeschreibung PID

Beschreibung der Bausteinparameter

Parameter	Datentyp	Bedeutung
SP	REAL	Führungsgröße
PV	REAL	Regelgröße
MODE	Mode_PID	Betriebsarten
PARA	Para_PID	Parameter
FEED_FWD	REAL	Störgröße
YMAN	REAL	Hand-Stellwert
ERR	REAL	Regeldifferenz
Y	REAL	Stellgröße
STATUS	Stat_MAXMIN	Status des Ausgangs Y

Parameterbeschreibung Mode_PID

Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
man	BOOL	"1": Betriebsart Hand
halt	BOOL	"1": Betriebsart Halt
en_p	BOOL	"1": P-Anteil ein
en_i	BOOL	"1": I-Anteil ein
en_d	BOOL	"1": D-Anteil ein
d_on_pv	BOOL	"1": D-Anteil bezogen auf Regelgröße; "0": D-Anteil bezogen auf Regeldifferenz

Parameterbeschreibung Para_PID

Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
gain	REAL	Proportionalbeiwert (Verstärkungsfaktor)
ti	TIME	Nachstellzeit
td	TIME	Vorhaltezeit
td_lag	TIME	Verzögerung des D-Anteils
ymax	REAL	Obere Stellgrenze
ymin	REAL	Untere Stellgrenze

Parameterbeschreibung Stat_MAXMIN

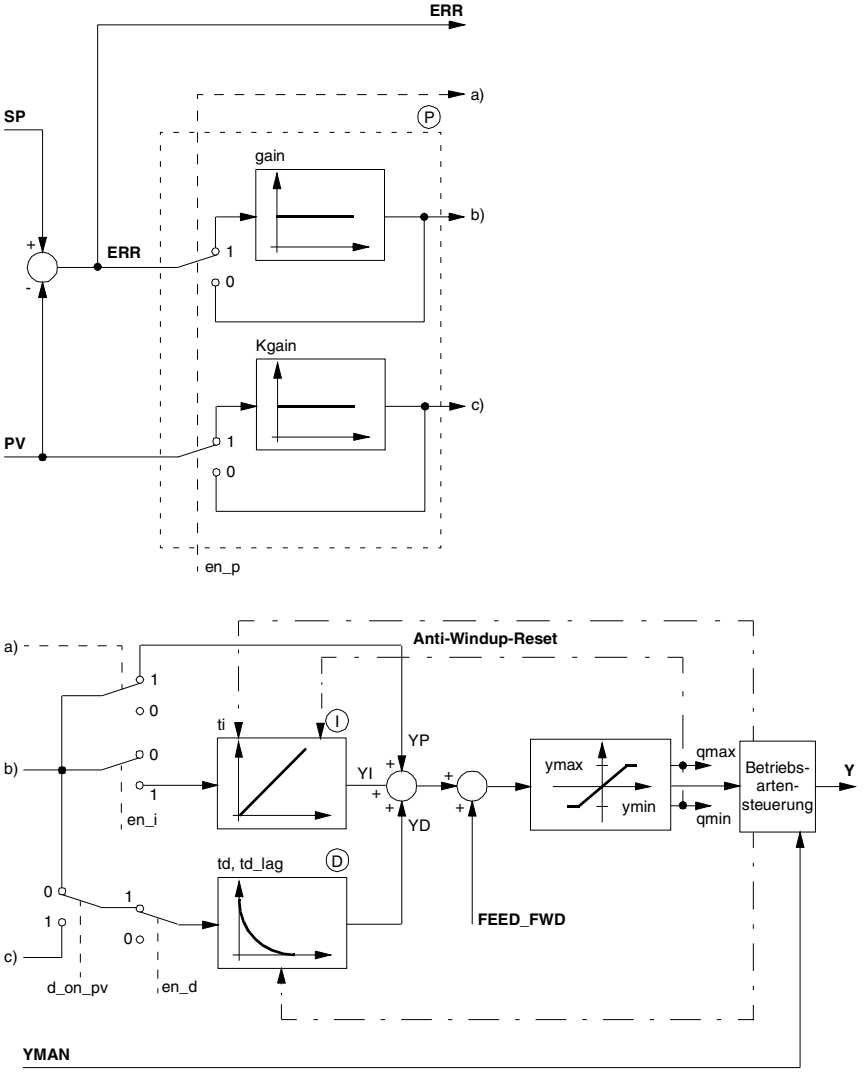
Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
qmax	BOOL	"1" = Y hat obere Stellgrenze erreicht
qmin	BOOL	"1" = Y hat untere Stellgrenze erreicht

Strukturbild des Funktionsbausteins PID

Strukturbild

Nachfolgend finden Sie das Strukturbild des Bausteins PID:



Parametrierung des PID-Reglers

Parametrierung

Die Struktur des PID-Reglers ist im *Strukturbild, S. 309* dargestellt.

Die Parametrierung des Funktionsbausteins erfolgt zunächst durch die reinen PID-Parameter, nämlich den Proportionalbeiwert gain, der Nachstellzeit ti und der Vorhaltezeit td.

Der D-Anteil wird mit der Zeit td_lag verzögert. Das Verhältnis von td/td_lag nennt man Differenzierverstärkung und wird im allgemeinen zwischen 3 und 10 gewählt. Der D-Anteil kann entweder basierend auf der Regeldifferenz ERR (d_on_pv = 0) oder basierend auf der Regelgröße PV (d_on_pv = 1) gebildet werden. Wird der D-Anteil aufgrund der Regelgröße PV bestimmt, so wird bei Führungsgrößenänderungen (Änderungen im Eingang SP) kein Sprung bedingt durch den D-Anteil entstehen. Der D-Anteil wirkt sich dann im Prinzip nur auf Störungen und Prozeßänderungen aus.

Umkehr des Regelsinns

Ein entgegengesetztes Verhalten des Reglers kann durch Umkehrung des Vorzeichens an gain erreicht werden. Ein positiver Wert an gain bewirkt, daß bei einer positiven Störgröße der Ausgangswert steigt. Ein negativer Wert an gain bewirkt, daß bei einer positiven Störgröße der Ausgangswert fällt.

Stellgrößen-Begrenzung

Die Grenzen ymax und ymin begrenzen den Ausgang nach oben sowie nach unten. Damit gilt ymin ≤ Y ≤ ymax.

Das Erreichen der Grenzwerte, bzw. eine Begrenzung des Ausgangssignals wird durch die Ausgänge qmax und qmin angezeigt:

- qmax = 1 wenn Y ≥ ymax
- qmin = 1 wenn Y ≤ ymin

Die Obergrenze ymax für die Begrenzung der Stellgröße ist größer als die Untergrenze ymin zu wählen, ansonsten meldet der Funktionsbaustein einen Fehler und arbeitet nicht.

Anti-Windup-Reset

Findet eine Stellgrößen-Begrenzung statt, soll durch Anti-Windup-Reset dafür gesorgt werden, daß der Integrale Anteil "nicht über alle Grenzen wachsen kann". Die Anti-Windup-Maßnahme wird nur dann durchgeführt, wenn der I-Anteil des Reglers nicht abgeschaltet ist. Die Grenzen für Anti-Windup sind hier dieselben, wie für die Stellgrößenbegrenzung. Der D-Anteil wird bei der Anti-Windup-Maßnahme nicht berücksichtigt, so daß Spitzen, hervorgerufen durch den D-Anteil, nicht durch die Anti-Windup-Maßnahme gekappt werden.

Die Anti-Windup-Maßnahme korrigiert den I-Anteil in der Form, daß gilt:

$$ymin - YP - FEED_FWD \leq YI \leq ymax - YP - FEED_FWD$$

Auswahl der Reglertypen

Es gibt verschiedene Reglertypen, die über die Elemente en_p, en_i und en_d ausgewählt werden:

Reglertyp	**en_p**	**en_i**	**en_d**
P-Regler	1	0	0
PI-Regler	1	1	0
PD-Regler	1	0	1
PID-Regler	1	1	1
I-Regler	0	1	0

Der I-Anteil kann auch mit ti = 0 abgeschaltet werden.

Betriebsarten

Auswahl der Betriebsarten

Es gibt drei Betriebsarten, die über die Parameter man und halt ausgewählt werden:

Betriebsart	man	halt
Automatik	0	0
Hand	1	0 oder 1
Halt	0	1

Betriebsart Automatik

In der Betriebsart Automatik wird die Stellgröße Y durch den diskretisierten PID-Algorithmus, in Abhängigkeit von Regelgröße PV und Führungsgröße SP bestimmt. Die Stellgröße wird durch ymax und ymin begrenzt. Die Stellgrenzen sind gleichzeitig auch Grenzen für den Antiwindup-Reset.

Betriebsart Hand

In der Betriebsart Hand wird der Handstellwert YMAN fest auf die Stellgröße Y übertragen. Die Stellgröße wird jedoch durch ymax und ymin begrenzt. Die internen Größen werden so nachgeführt, daß der Regler (bei zugeschaltetem I-Anteil) stoßfrei von Hand auf Automatik umgeschaltet werden kann. Die Stellgrenzen sind gleichzeitig auch Grenzen für den Antiwindup-Reset.

Der D-Anteil wird in dieser Betriebsart automatisch auf 0 gesetzt.

Betriebsart Halt

In der Betriebsart Halt bleibt der Reglerausgang, wie er vorgefunden wird, der Funktionsbaustein verändert die Stellgröße Y nicht (Regler bleibt stehen), d. h. Y = Y(old). Die internen Größen werden so nachgeführt, so daß der Regler (bei zugeschaltetem I-Anteil) ohne Stoß von seiner aktuellen Lage aus weiterfährt. Die Stellgrenzen sind gleichzeitig auch Grenzen für den Antiwindup-Reset. Die Betriebsart Halt eignet sich unter anderem auch dafür, den Reglerausgang Y über ein externes Bediengerät zu stellen, wobei die internen Anteile im Regler richtig nachgeführt werden.

Der D-Anteil wird in dieser Betriebsart automatisch auf 0 gesetzt.

Umschaltung von Automatik nach Hand

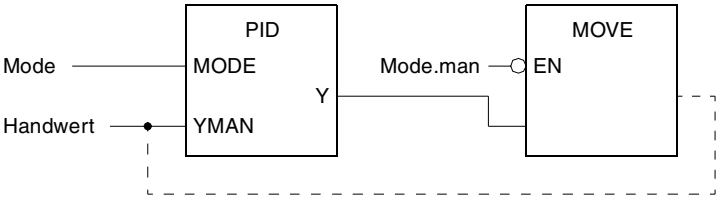
Die Umschaltung von Automatik nach Hand ist normalerweise nicht stoßfrei, da der Ausgang Y jeden Wert zwischen ymax und ymin haben kann und Y bei der Umschaltung direkt den Wert YMAN annimmt.

Soll die Umschaltung von Automatik nach Hand trotzdem stoßfrei erfolgen, gibt es zwei Möglichkeiten:

- Umschaltung mit Hilfe der Funktion MOVE
- Umschaltung mit Hilfe eines Anstiegsbegrenzer Funktionsbaustein VLIM

Umschaltung mit MOVE

Setzen Sie mit Hilfe der Funktion MOVE den Wert von YMAN auf den Wert von Y:

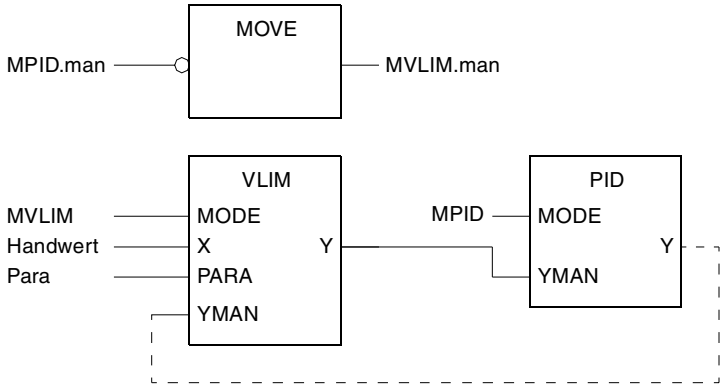


Hinweis: Diese Art der Darstellung wurde lediglich zur verständlicheren Darstellung gewählt. Die gestrichelt dargestellten Verbindungen können nicht als Links (Link-Objekte) programmiert werden, da sie (in Concept) nicht zulässige Schleifen bilden. Die Verbindungen müssen bei der Programmierung durch Variablen realisiert werden.

Die Funktion MOVE wird nur ausgeführt wenn sich der PID-Regler in der Betriebsart Automatik befindet (Mode.man = 0). Wenn nun eine Umschaltung von Automatik nach Hand erfolgt geschieht dies stoßfrei, da in diesem Zyklus der Wert von YMAN gleich dem Wert von Y ist. In der Betriebsart Hand können Sie nun den Wert von YMAN langsam verändern.

Umschaltung mit VLIM

Wenn Sie den Wert YMAN nicht ändern wollen, da er z.B. ein Festwert ist, können Sie die vorherige Lösung mit Hilfe eines Anstiegsbegrenzer (Funktionsbaustein VLIM) realisieren:



Hinweis: Diese Art der Darstellung wurde lediglich zur verständlicheren Darstellung gewählt. Die gestrichelt dargestellten Verbindungen können nicht als Links (Link-Objekte) programmiert werden, da sie (in Concept) nicht zulässige Schleifen bilden. Die Verbindungen müssen bei der Programmierung durch Variablen realisiert werden.

In der Betriebsart Automatik (MPID.man = 0) ist der Anstiegsbegrenzer in der Betriebsart Hand (MOVE-Funktion). Somit wird der Handwert des PID-Reglers (YMAN von PID) über den Handwert des Anstiegsbegrenzers (YMAN von VLIM) auf den Wert Y des PID-Reglers gesetzt (nachgeführt). Wenn nun eine Umschaltung von Automatik nach Hand erfolgt geschieht dies stoßfrei, da in diesem Zyklus der Wert YMAN (des PID) gleich dem Wert von Y (des PID) ist. Ab dem nächsten Zyklus wird der Wert von YMAN (des PID) mit der von Ihnen bestimmten Anpassung (Para.rate) an den eigentlichen Handwert (an VLIM) angeglichen.

Detaillierte Formeln

Erläuterung der Formelgrößen

Bedeutung der Größen in den nachfolgenden Formeln:

Größe	Bedeutung
dt	Zeitdifferenz zwischen aktullem Zyklus und vorherigem Zyklus
ERR	Regeldifferenz (SP - PV)
$ERR_{(new)}$	Wert der Regeldifferenz aus dem aktuellen Abtastschritt
$ERR_{(old)}$	Wert der Regeldifferenz aus dem vorherigen Abtastschritt
FEED_FWD	Störgröße
$PV_{(new)}$	Wert der Regelgröße aus dem aktuellen Abtastschritt
$PV_{(old)}$	Wert der Regelgröße aus dem vorherigen Abtastschritt.
Y	aktueller Ausgang (Betriebsart Halt) oder YMAN (Betriebsart Hand)
YD	D-Anteil
YI	I-Anteil
YP	P-Anteil

Stellgröße

Die Stellgröße setzt sich aus verschiedenen von den Betriebsarten abhängigen Teilgrößen zusammen:

$$Y = YP + YI + YD + FEED_FWD$$

Nach der Summation der Anteile findet eine Stellgrößen-Begrenzung statt, so daß gilt:

$$ymin \leq Y \leq ymax$$

Übersicht zur Berechnung der Regler-Anteile

Nachfolgend finden Sie eine Übersicht über die unterschiedliche Berechnung der Regler-Anteile in Abhängigkeit von den Elementen en_p, en_i und en_d:

- P-Anteil YP für Betriebsart Hand, Halt und Automatik
- I-Anteil YI für Betriebsart Automatik
- I-Anteil YI für Betriebsart Hand und Halt
- D-Anteil YD für Betriebsart Automatik
- D-Anteil YD für Betriebsart Hand und Halt

P-Anteil YP für alle Betriebsarten

YP für Betriebsart Hand, Halt und Automatik wird wie folgt ermittelt:

Für en_p = 1 gilt:

$$YP = gain \times ERR$$

Für en_p = 0 gilt:

$$YP = 0$$

I-Anteil YI für Betriebsart Automatik

YI für Betriebsart Automatik wird wie folgt ermittelt:

Für en_i = 1 gilt:

$$YI_{(new)} = YI_{(old)} + gain \times \frac{dt}{ti} \times \frac{ERR_{(new)} + ERR_{(old)}}{2}$$

Für en_i = 0 gilt:

$$YI = 0$$

Der I-Anteil wird nach der Trapezregel gebildet.

I-Anteil YI für Betriebsart Hand und Halt

YI für Betriebsart Hand und Halt wird wie folgt ermittelt:

Für en_i = 1 gilt:

$$YI = Y - (YP - FEED_FWD)$$

Für en_i = 0 gilt:

$$YI = 0$$

D-Anteil YD für Betriebsart Automatik

YD für Betriebsart Automatik und Kaskade wird wie folgt ermittelt:

Für en_d = 1 und d_on_pv = 0 gilt:

$$YD_{(new)} = \frac{YD_{(old)} \times td_lag + td \times gain \times (ERR_{(new)} - ERR_{(old)})}{dt + dt_lag}$$

Für en_d = 1 und d_on_pv = 1 gilt:

$$YD_{(new)} = \frac{YD_{(old)} \times td_lag + td \times gain \times (PV_{(old)} - PV_{(new)})}{dt + dt_lag}$$

Für en_d = 0 gilt:

$$YD = 0$$

D-Anteil YD für Betriebsart Hand und Halt

YD für Betriebsart Hand und Halt wird wie folgt ermittelt:

YD = 0

Laufzeitfehler

Fehlermeldung

Es erfolgt eine Fehlermeldung, wenn

- eine unzulässige Gleitkommazahl am Eingang YMAN oder PV anliegt
- ymax < ymin ist

PID1: PID-Regler

36

Übersicht

Einleitung

Diese Kapitel beschreibt den Baustein PID1.

Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Kurzbeschreibung	320
Darstellung	321
Strukturbild des Funktionsbausteins PID1	323
Parametrierung des PID1-Reglers	324
Betriebsarten	326
Detaillierte Formeln	329
Laufzeitfehler	330

Kurzbeschreibung

Funktionsbeschreibung

Der Funktionsbaustein realisiert einen PID-Regler.

Aufgrund der Führungsgröße SP und der Regelgröße PV wird eine Regeldifferenz ERR gebildet. Diese Regeldifferenz ERR bewirkt eine Änderung der Stellgröße Y.

Als zusätzliche Parameter können EN und ENO projektiert werden.

Eigenschaften

Der Funktionsbaustein besitzt folgenden Eigenschaften:

- realer PID-Regler mit unabhängiger GAIN, TI, TD-Einstellung
- Betriebsarten Hand, Halt, Automatik
- stoßfreie Umschaltung Hand/Automatik
- Stellgrößen-Begrenzung im Automatik-Betrieb
- Einzeln zuschaltbarer P-, I- und D-Anteil
- Anti-Windup-Reset
- Anti-Windup-Maßnahme nur bei aktivem I-Anteil
- definierbare Verzögerung des D-Anteils
- D-Anteil umschaltbar auf Regelgröße PV oder Regeldifferenz ERR

Übertragungsfunktion

Die Übertragungsfunktion lautet:

$$G(s) = GAIN \times \left(1 + \frac{1}{TI \times s} + \frac{TD \times s}{1 + TD_LAG \times s}\right)$$

YD
YI
YP

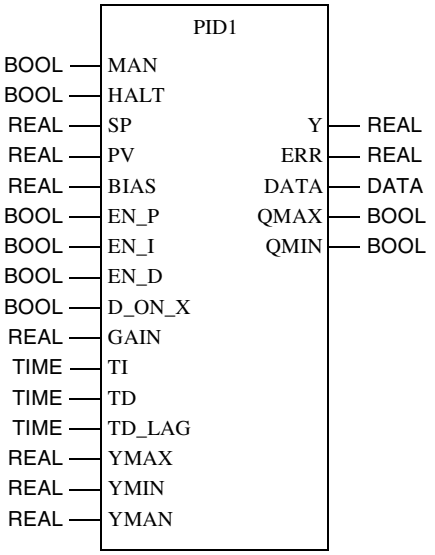
Erläuterung der Größen

Größe	Bedeutung
YD	D-Anteil (nur bei EN_D = 1)
YI	I-Anteil (nur bei EN_I = 1)
YP	P-Anteil (nur bei EN_P = 1)

Darstellung

Symbol

Darstellung des Bausteins:



Parameterbeschreibung

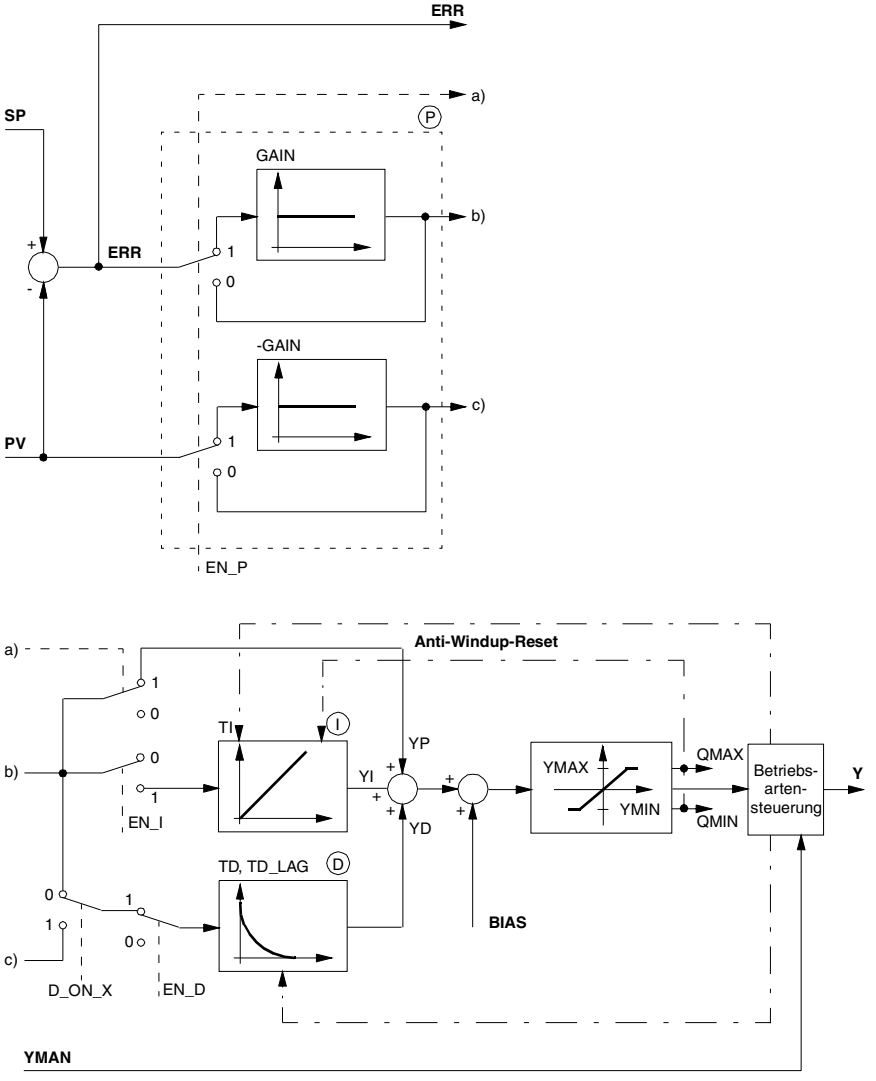
Beschreibung der Bausteinparameter

Parameter	Datentyp	Bedeutung
MAN	BOOL	"1": Betriebsart Hand
HALT	BOOL	"1": Betriebsart Halt
SP	REAL	Sollwerteingang
PV	REAL	Regelgröße (Process Variable)
BIAS	REAL	Störgrößeneingang
EN_P	BOOL	"1": P-Anteil ein
EN_I	BOOL	"1": I-Anteil ein
EN_D	BOOL	"1": D-Anteil ein
D_ON_X	BOOL	"1": D-Anteil auf Regelgröße; "0": D-Anteil auf Regeldifferenz
GAIN	REAL	Proportionalbeiwert (Verstärkungsfaktor)
TI	TIME	Nachstellzeit
TD	TIME	Vorhaltezeit
TD_LAG	TIME	Verzögerungszeit, D-Anteil
YMAX	REAL	Obere Stellgrenze
YMIN	REAL	Untere Stellgrenze
YMAN	REAL	Hand-Stellwert
ERR	REAL	Ausgang Regeldifferenz
Y	REAL	Stellgröße
QMAX	BOOL	"1" = Ausgang Y hat obere Stellgrenze erreicht
QMIN	BOOL	"1" = Ausgang Y hat untere Stellgrenze erreicht

Strukturbild des Funktionsbausteins PID1

Strukturbild

Nachfolgend finden Sie das Strukturbild des Bausteins PID1:



Parametrierung des PID1-Reglers

Parametrierung

Die Struktur des PID1-Reglers ist im *Strukturbild, S. 323* dargestellt.

Die Parametrierung des Funktionsbausteins erfolgt zunächst durch die reinen PID-Parameter, nämlich den Proportionalbeiwert GAIN, der Nachstellzeit TI und der Vorhaltezeit TD.

Der D-Anteil wird mit der Zeit TD_LAG verzögert. Das Verhältnis von TD/TD_LAG nennt man Differenzierverstärkung VD. Der D-Anteil kann entweder durch die Regeldifferenz ERR (D_ON_X = 0) oder durch die Regelgröße PV (D_ON_X = 1) gebildet werden. Wird der D-Anteil durch die Regelgröße PV bestimmt, so wird bei Führungsgrößenänderungen (Änderungen im Eingang SP) kein Sprung bedingt durch den D-Anteil entstehen. Der D-Anteil wirkt sich im Prinzip nur auf Störungen und Prozeßänderungen aus.

Umkehr des Regelsinns

Ein entgegengesetztes Verhalten des Reglers kann durch Umkehrung des Vorzeichens an GAIN erreicht werden. Ein positiver Wert an GAIN bewirkt, daß bei einer positiven Störgröße der Ausgangswert steigt. Ein negativer Wert an GAIN bewirkt, daß bei einer positiven Störgröße der Ausgangswert fällt.

Stellgrößen-Begrenzung

Die Grenzen YMAX und YMIN begrenzen den Ausgang nach oben sowie nach unten. Damit gilt YMIN ≤ Y ≤ YMAX.

Das Erreichen der Grenzwerte, bzw. eine Begrenzung des Ausgangssignals wird durch die beiden Merker QMAX und QMIN angezeigt:

- QMAX = 1 wenn Y ≥ YMAX
- QMIN = 1 wenn Y ≤ YMIN

Die Obergrenze YMAX für die Begrenzung der Stellgröße ist größer als die Untergrenze YMIN zu wählen.

Anti-Windup-Reset

Findet eine Stellgrößen-Begrenzung statt, soll durch Anti-Windup-Reset dafür gesorgt werden, daß der Integrale Anteil "nicht über alle Grenzen wachsen kann". Die Anti-Windup-Maßnahme wird nur dann durchgeführt, wenn der I-Anteil des Reglers nicht abgeschaltet ist. Die Grenzen für Anti-Windup sind hier dieselben, wie für die Stellgrößenbegrenzung. Der D-Anteil wird für der Anti-Windup-Maßnahme nicht mit berücksichtigt, so daß Spitzen, hervorgerufen durch den D-Anteil, nicht durch die Anti-Windup-Maßnahme gekappt werden.

Die Anti-Windup-Maßnahme korrigiert den I-Anteil in der Form, daß gilt:

$$YMIN - YP - BIAS \leq YI \leq YMAX - YP - BIAS$$

Auswahl der Reglertypen

Es gibt verschiedene Reglertypen, die über die Parameter EN_P, EN_I und EN_D ausgewählt werden:

Reglertyp	EN_P	EN_I	EN_D
P-Regler	1	0	0
PI-Regler	1	1	0
PD-Regler	1	0	1
PID-Regler	1	1	1
I-Regler	0	1	0

Der I-Anteil kann auch mit TI = 0 abgeschaltet werden.

Betriebsarten

Auswahl der Betriebsarten

Es gibt drei Betriebsarten, die über die Parameter MAN und HALT ausgewählt werden:

Betriebsart	MAN	HALT
Automatik	0	0
Hand	1	0 oder 1
Halt	0	1

Betriebsart Automatik

In der Betriebsart Automatik wird die Stellgröße Y durch den diskretisierten PID-Algorithmus, in Abhängigkeit von Regelgröße PV und Führungsgröße SP bestimmt. Die Stellgröße wird durch ymax und ymin begrenzt. Die Stellgrenzen sind gleichzeitig auch Grenzen für den Antiwindup-Reset.

Betriebsart Hand

In der Betriebsart Hand wird der Handstellwert YMAN fest auf den Stellausgang Y übertragen. Der Stellausgang wird jedoch durch YMAX und YMIN begrenzt. Die internen Größen werden so nachgeführt, daß der Regler (bei zugeschaltetem I-Anteil) stoßfrei von Hand auf Automatik umgeschaltet werden kann. Die Stellgrenzen sind gleichzeitig auch Grenzen für den Antiwindup-Reset.

Der D-Anteil wird in dieser Betriebsart automatisch auf 0 gesetzt.

Betriebsart Halt

In der Betriebsart Halt bleibt der Reglerausgang, wie er vorgefunden wird, der Funktionsbaustein verändert den Reglerausgang Y nicht (Regler bleibt stehen), d. h. Y = Y(old). Die internen Größen werden so nachgeführt, daß die Summe ihrer Anteile dem Stellausgang entsprechen, so daß der Regler (bei zugeschaltetem I-Anteil) ohne Stoß von seiner aktuellen Lage aus weiterfährt. Die Stellgrenzen sind gleichzeitig auch Grenzen für den Antiwindup-Reset.

Der D-Anteil wird in dieser Betriebsart automatisch auf 0 gesetzt.

Umschaltung von Automatik nach Hand

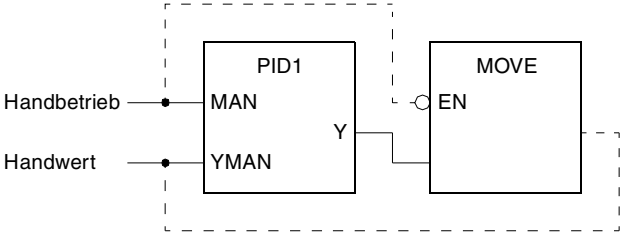
Die Umschaltung von Automatik nach Hand ist normalerweise nicht stoßfrei, da der Ausgang Y jeden Wert zwischen ymax und ymin haben kann und Y bei der Umschaltung direkt den Wert YMAN annimmt.

Soll die Umschaltung von Automatik nach Hand trotzdem stoßfrei erfolgen, gibt es zwei Möglichkeiten:

- Umschaltung mit Hilfe der Funktion MOVE
- Umschaltung mit Hilfe eines Anstiegsbegrenzer Funktionsbaustein LIMV

Umschaltung mit MOVE

Setzen Sie mit Hilfe der Funktion MOVE den Wert von YMAN auf den Wert von Y:

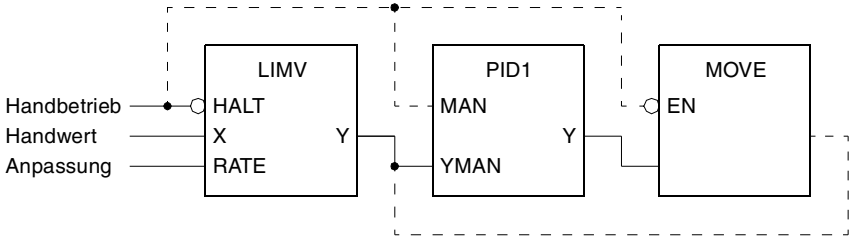


Hinweis: Diese Art der Darstellung wurde lediglich zur verständlicheren Darstellung gewählt. Die gestrichelt dargestellten Verbindungen können nicht als Links (Link-Objekte) programmiert werden, da sie (in Concept) nicht zulässige Schleifen bilden. Die Verbindungen müssen bei der Programmierung durch Variablen realisiert werden.

Die Funktion MOVE wird nur ausgeführt wenn sich der PID-Regler in der Betriebsart Automatik befindet (MAN = 0). Wenn nun eine Umschaltung von Automatik nach Hand erfolgt geschieht dies stoßfrei, da in diesem Zyklus der Wert von YMAN gleich dem Wert von Y ist. In der Betriebsart Hand können Sie nun den Wert von YMAN langsam verändern.

Umschaltung mit LIMV

Wenn Sie den Wert YMAN nicht ändern wollen, da er z.B. ein Festwert ist, müssen Sie die vorherige Lösung durch einen Anstiegsbegrenzer (Funktionsbaustein LIMV (siehe *LIMV: Anstiegsbegrenzer 1. Ordnung, S. 207*)) ergänzen:



Hinweis: Diese Art der Darstellung wurde lediglich zur verständlicheren Darstellung gewählt. Die gestrichelt dargestellten Verbindungen können nicht als Links (Link-Objekte) programmiert werden, da sie (in Concept) nicht zulässige Schleifen bilden. Die Verbindungen müssen bei der Programmierung durch Variablen realisiert werden.

Die Funktion MOVE wird nur ausgeführt wenn sich der PID-Regler in der Betriebsart Automatik befindet (MAN = 0). Wenn nun eine Umschaltung von Automatik nach Hand erfolgt geschieht dies stoßfrei, da in diesem Zyklus der Wert YMAN (des PID1) gleich dem Wert von Y (des PID1) ist. Ab dem nächsten Zyklus wird der Wert von YMAN (des PID1) mit der von Ihnen bestimmten Anpassung (RATE) an den eigentlichen Handwert (an LIMV) angeglichen.

Detaillierte Formeln

Erläuterung der Formelgrößen

Bedeutung der Größen in den nachfolgenden Formeln:

Größe	Bedeutung
dt	Zeitdifferenz zwischen aktullem Zyklus und vorherigem Zyklus
ERR	Regeldifferenz (SP - PV)
$ERR_{(new)}$	Wert der Regeldifferenz aus dem aktuellen Abtastschritt
$ERR_{(old)}$	Wert der Regeldifferenz aus dem vorherigen Abtastschritt
BIAS	Störgröße
$PV_{(new)}$	Wert der Regelgröße aus dem aktuellen Abtastschritt
$PV_{(old)}$	Wert der Regelgröße aus dem vorherigen Abtastschritt.
Y	aktueller Ausgang (Betriebsart Halt) oder YMAN (Betriebsart Hand)
YD	D-Anteil
YI	I-Anteil
YP	P-Anteil

Stellgröße

Die Stellgröße setzt sich aus verschiedenen von den Betriebsarten abhängigen Teilgrößen zusammen:

$$Y = YP + YI + YD + BIAS$$

Nach der Summation der Anteile findet eine Stellgrößen-Begrenzung statt, so daß gilt:

$$YMIN \leq Y \leq YMAX$$

Übersicht zur Berechnung der Regler-Anteile

Nachfolgend finden Sie eine Übersicht über die unterschiedliche Berechnung der Regler-Anteile in Abhängigkeit von den Eingängen EN_P, EN_I und EN_D:

- P-Anteil YP für Betriebsart Hand, Halt und Automatik
- I-Anteil YI für Betriebsart Automatik
- I-Anteil YI für Betriebsart Hand und Halt
- D-Anteil YD für Betriebsart Automatik
- D-Anteil YD für Betriebsart Hand und Halt

P-Anteil YP für alle Betriebsarten

YP für Betriebsart Hand, Halt und Automatik wird wie folgt ermittelt:

Für EN_P = 1 gilt:

$$YP = GAIN \times ERR$$

Für EN_P = 0 gilt:

$$YP = 0$$

I-Anteil YI für Betriebsart Automatik

YI für Betriebsart Automatik wird wie folgt ermittelt:

Für EN_I = 1 gilt:

$$YI_{(new)} = YI_{(old)} + GAIN \times \frac{dt}{TI} \times \frac{ERR_{(new)} + ERR_{(old)}}{2}$$

Für EN_I = 0 gilt:

$$YI = 0$$

Der I-Anteil wird nach der Trapezregel gebildet.

I-Anteil YI für Betriebsart Hand und Halt

YI für Betriebsart Hand und Halt wird wie folgt ermittelt:

Für EN_I = 1 gilt:

$$YI = Y - (YP - BIAS)$$

Für EN_I = 0 gilt:

$$YI = 0$$

D-Anteil YD für Betriebsart Automatik

YD für Betriebsart Automatik und Kaskade wird wie folgt ermittelt:

Für EN_D = 1 und D_ON_X = 0 gilt:

$$YD_{(new)} = \frac{YD_{(old)} \times TD_LAG + TD \times GAIN \times (ERR_{(new)} - ERR_{(old)})}{dt + TD_LAG}$$

Für EN_D = 1 und D_ON_X = 1 gilt:

$$YD_{(new)} = \frac{YD_{(old)} \times TD_LAG + TD \times GAIN \times (PV_{(old)} - PV_{(new)})}{dt + TD_LAG}$$

Für EN_D = 0 gilt:

$$YD = 0$$

D-Anteil YD für Betriebsart Hand und Halt

YD für Betriebsart Hand und Halt wird wie folgt ermittelt:

YD = 0

Laufzeitfehler

Fehlermeldung

Bei YMAX < YMIN erfolgt eine Fehlermeldung.

PID_P: PID-Regler mit paralleler Struktur

37

Übersicht

Einleitung

Diese Kapitel beschreibt den Baustein PID_P.

Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Kurzbeschreibung	332
Darstellung	333
Parametrierung des PID_P-Reglers	335
Betriebsarten	337
Detaillierte Formeln	338
Laufzeitfehler	339

Kurzbeschreibung

Funktionsbeschreibung

Der Funktionsbaustein realisiert einen PID-Regler in reiner Parallel-Struktur.

Zwischen Führungsgröße SP und der Regelgröße PV wird die Regeldifferenz ERR gebildet. Die Regeldifferenz ERR bewirkt eine Änderung der Stellgröße Y.

Als zusätzliche Parameter können EN und ENO projektiert werden.

Eigenschaften

Der Funktionsbaustein besitzt folgenden Eigenschaften:

- PID-Regler in reiner Parallel-Struktur
- Unabhängige Verstärkungsfaktoren für P-. I- und D-Anteil
- Jeder Anteil P, I oder D kann einzeln abgeschaltet werden
- Stellgrenzen-Begrenzung im Automatik-Betrieb
- Antiwindup-Maßnahme nur bei aktivem I-Anteil
- Anti-Windup-Reset
- Betriebsarten Hand, Halt, Automatik
- stoßfreie Umschaltung Hand/Automatik
- D-Anteil umschaltbar auf Eingangsgröße PV bzw. Regeldifferenz ERR
- D-Anteil mit variabler Zeitverzögerung

Übertragungsfunktion

Die Übertragungsfunktion lautet:

$$G(s) = kp + \frac{ki}{s} + \frac{kd \times s}{s + \frac{1}{td_lag}}$$

YD
YI
YP

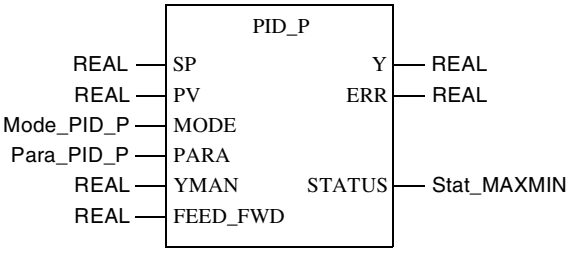
Erläuterung der Größen:

Größe	Bedeutung
YD	D-Anteil
YI	I-Anteil
YP	P-Anteil

Darstellung

Symbol

Darstellung des Bausteins:



Parameterbeschreibung PID_P

Beschreibung der Bausteinparameter

Parameter	Datentyp	Bedeutung
SP	REAL	Führungsgröße
PV	REAL	Regelgröße
MODE	Mode_PID_P	Betriebsarten
PARA	Para_PID_P	Parameter
YMAN	REAL	Hand-Stellwert
FEED_FWD	REAL	Störgrößeneingang
Y	REAL	Stellgröße
ERR	REAL	Regeldifferenz
STATUS	Stat_MAXMIN	Status des Ausgangs Y

Parameterbeschreibung Mode_PID_P

Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
man	BOOL	"1": Betriebsart Hand
halt	BOOL	"1": Betriebsart Halt
d_on_pv	BOOL	"1": D-Anteil bezogen auf Regelgröße; "0": D-Anteil bezogen auf Regeldifferenz
reverse	BOOL	"1": Ausgang umgekehrt

Parameterbeschreibung Para_PID_P

Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
kp	REAL	Proportionalbeiwert (Verstärkungsfaktor = P-Anteil)
ki	REAL	Integrierbeiwert (Verstärkungsfaktor I-Anteil) [1/s]
kd	REAL	Differenzierbeiwert (Verstärkungsfaktor D-Anteil) [s]
td_lag	TIME	Verzögerungszeit (Einheit = s), D-Anteil
ymax	REAL	Obere Stellgrenze
ymin	REAL	Untere Stellgrenze

Parameterbeschreibung Stat_MAXMIN

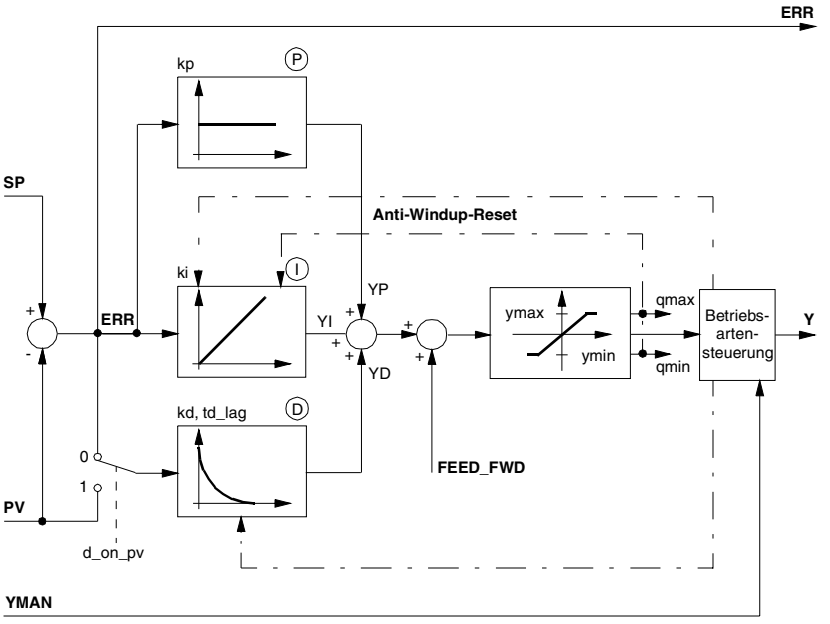
Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
qmax	BOOL	"1" = Y hat obere Stellgrenze erreicht
qmin	BOOL	"1" = Y hat untere Stellgrenze erreicht

Parametrierung des PID_P-Reglers

Strukturbild

Nachfolgend finden Sie das Strukturbild des Bausteins PID_P:



Parametrierung

Die Struktur des PID_P-Reglers ist im *Strukturbild, S. 335* dargestellt.

Die Parametrierung des PID_P-Reglers erfolgt zunächst durch die reinen PID-Parameter, nämlich den Proportionalbeiwert kp, den Integrierbeiwert ki und den Differenzierbeiwert kd.

Der P-, I- und D-Anteil können einzeln abgeschaltet werden, indem der entsprechende Eingang (kp, ki oder kd) auf 0 gesetzt wird.

Der D-Anteil wird mit der Zeitkonstante td_lag verzögert. Der D-Anteil kann entweder basierend auf der Regeldifferenz ERR (d_on_pv =0) oder basierend auf der Regelgröße PV (d_on_pv = 1) gebildet werden. Wird der D-Anteil aufgrund der Regelgröße PV bestimmt, so wird bei Führungsgrößenänderungen (Änderungen im Eingang SP) kein Sprung bedingt durch den D-Anteil entstehen. Der D-Anteil wirkt sich im Prinzip nur auf Störungen bzw. Prozeßänderungen aus.

Umkehr des Regelsinns

Ein entgegengesetztes Verhalten des Reglers kann durch Setzen des Eingangs reverse auf 1 erreicht werden. reverse = 0 bewirkt, daß bei einer positiven Störgröße der Ausgangswert steigt. reverse = 1 bewirkt, daß bei einer positiven Störgröße der Ausgangswert fällt.

Stellgrößen-Begrenzung

Die Grenzen ymax und ymin begrenzen den Ausgang nach oben sowie nach unten. Damit gilt ymin ≤ Y ≤ ymax.

Das Erreichen der Grenzwerte, bzw. eine Begrenzung des Ausgangssignals wird durch die Ausgänge qmax und qmin angezeigt:

- qmax = 1 wenn Y ≥ ymax
- qmin = 1 wenn Y ≤ ymin

Die Obergrenze ymax für die Begrenzung der Stellgröße ist größer als die Untergrenze ymin zu wählen, ansonsten meldet der Funktionsbaustein einen Fehler und arbeitet nicht.

Anti-Windup-Reset

Findet eine Stellgrößen-Begrenzung statt, soll durch Anti-Windup-Reset dafür gesorgt werden, daß der Integrale Anteil "nicht über alle Grenzen wachsen kann". Die Anti-Windup-Maßnahme wird nur dann durchgeführt, wenn der I-Anteil des Reglers nicht abgeschaltet ist. Die Grenzen für Anti-Windup sind hier dieselben, wie für die Stellgrößenbegrenzung. Der D-Anteil wird bei der Anti-Windup-Maßnahme nicht berücksichtigt, so daß Spitzen, hervorgerufen durch den D-Anteil, nicht durch die Anti-Windup-Maßnahme gekappt werden.

Die Anti-Windup-Maßnahme korrigiert den I-Anteil in der Form, daß gilt:

$$\mathrm{ymin} - \mathrm{YP} - \mathrm{FEED_FWD} \leq \mathrm{YI} \leq \mathrm{ymax} - \mathrm{YP} - \mathrm{FEED_FWD}$$

Auswahl der Reglertypen

Es gibt verschiedene Reglertypen, die über die Parameter kp, ki und kd ausgewählt werden:

Reglertyp	kp	ki	kd
P-Regler	> 0	= 0	= 0
PI-Regler	> 0	> 0	= 0
PD-Regler	> 0	= 0	> 0
PID-Regler	> 0	> 0	> 0
I-Regler	= 0	> 0	= 0

Betriebsarten

Auswahl der Betriebsarten

Es gibt drei Betriebsarten, die über die Parameter man und halt ausgewählt werden:

Betriebsart	man	halt
Automatik	0	0
Hand	1	0 oder 1
Halt	0	1

Betriebsart Automatik

In der Betriebsart Automatik wird die Stellgröße Y durch den diskretisierten PID-Algorithmus, in Abhängigkeit von Regelgröße PV und Führungsgröße SP bestimmt. Die Stellgröße wird durch ymax und ymin begrenzt. Die Stellgrenzen sind gleichzeitig auch Grenzen für den Antiwindup-Reset.

Die Umschaltung von Automatik nach Hand ist normalerweise nicht stoßfrei, da der Ausgang Y jeden Wert zwischen ymax und ymin haben kann und Y bei der Umschaltung direkt den Wert YMAN annimmt.

Soll die Umschaltung von Automatik nach Hand trotzdem stoßfrei erfolgen, gibt es zwei Möglichkeiten, die für einen PID-Regler (siehe *Umschaltung von Automatik nach Hand, S. 312*) beispielhaft erklärt sind.

Betriebsart Hand

In der Betriebsart Hand wird der Handstellwert YMAN fest auf die Stellgröße Y übertragen. Die Stellgröße wird jedoch durch ymax und ymin begrenzt. Die internen Größen werden so nachgeführt, daß der Regler (bei zugeschaltetem I-Anteil) stoßfrei von Hand auf Automatik umgeschaltet werden kann. Die Stellgrenzen sind gleichzeitig auch Grenzen für den Antiwindup-Reset.

Der D-Anteil wird in dieser Betriebsart automatisch auf 0 gesetzt.

Betriebsart Halt

In der Betriebsart Halt bleibt der Reglerausgang, wie er vorgefunden wird, der Funktionsbaustein verändert die Stellgröße Y nicht (Regler bleibt stehen), d. h. Y = Y(old). Die internen Größen werden so nachgeführt, daß der Regler (bei zugeschaltetem I-Anteil) ohne Stoß von seiner aktuellen Lage aus weiterfährt. Die Stellgrenzen sind gleichzeitig auch Grenzen für den Antiwindup-Reset. Die Betriebsart Halt eignet sich unter anderem auch dafür, den Reglerausgang Y über ein externes Bediengerät zu stellen, wobei die internen Anteile im Regler richtig nachgeführt werden.

Der D-Anteil wird in dieser Betriebsart automatisch auf 0 gesetzt.

Detaillierte Formeln

Erläuterung der Formelgrößen

Bedeutung der Größen in den Formeln:

Größe	Bedeutung
dt	Zeitdifferenz zwischen aktullem Zyklus und vorherigem Zyklus
ERR	Regeldifferenz (SP - PV)
$ERR_{(new)}$	Wert der Regeldifferenz aus dem aktuellen Abtastschritt
$ERR_{(old)}$	Wert der Regeldifferenz aus dem vorherigen Abtastschritt
FEED_FWD	Störgröße
$PV_{(new)}$	Wert der Regelgröße aus dem aktuellen Abtastschritt
$PV_{(old)}$	Wert der Regelgröße aus dem vorherigen Abtastschritt.
Y	aktueller Ausgang (Betriebsart Halt) oder YMAN (Betriebsart Hand)
YD	D-Anteil
YI	I-Anteil
YP	P-Anteil

Stellgröße

Die Stellgröße setzt sich aus verschiedenen Teilgrößen zusammen:
$Y = YP + YI + YD + FEED_FWD$

Nach der Summation der Anteile findet eine Stellgrößen-Begrenzung statt, so daß gilt:

$ymin \leq Y \leq ymax$

Regeldifferenz

Die Regeldifferenz wird wie folgt ermittelt:

ERR = SP - PV, wenn reverse = 0

ERR = PV - SP, wenn reverse = 1

Übersicht zur Berechnung der Regler-Anteile

Nachfolgend finden Sie eine Übersicht über die unterschiedliche Berechnung der Regler-Anteile in Abhängigkeit von den Verstärkungsfaktoren kp, ki und kd:

- P-Anteil YP für Betriebsart Hand, Halt und Automatik
- I-Anteil YI für Betriebsart Automatik
- I-Anteil YI für Betriebsart Hand und Halt
- D-Anteil YD für Betriebsart Automatik
- D-Anteil YD für Betriebsart Hand und Halt

P-Anteil YP für alle Betriebsarten

YP für Betriebsart Hand, Halt und Automatik wird wie folgt ermittelt:

$$YP = kp \times ERR$$

I-Anteil YI für Betriebsart Automatik

YI für Betriebsart Automatik wird wie folgt ermittelt:

Für ki > 0 gilt:

$$YI_{(new)} = YI_{(old)} + ki \times dt \times \frac{ERR_{(new)} + ERR_{(old)}}{2}$$

Für ki = 0 gilt:

$$YI = 0$$

Der I-Anteil wird nach der Trapezregel gebildet.

I-Anteil YI für Betriebsart Hand und Halt

YI für Betriebsart Hand und Halt wird wie folgt ermittelt:

Für ki > 0 gilt:

$$YI = Y - (YP - FEED_FWD)$$

Für ki = 0 gilt:

$$YI = 0$$

D-Anteil YD für Betriebsart Automatik

YD für Betriebsart Automatik und Kaskade wird wie folgt ermittelt:

Für kd > 0 und d_on_pv = 0 gilt:

$$YD_{(new)} = \frac{td_lag}{dt + td_lag} \times (YD_{(old)} + kd \times (ERR_{(new)} - ERR_{(old)}))$$

Für kd > 0 und d_on_pv = 1 gilt:

$$YD_{(new)} = \frac{td_lag}{dt + td_lag} \times (YD_{(old)} + kd \times (PV_{(old)} - PV_{(new)}))$$

Für kd = 0 gilt:

$$YD = 0$$

D-Anteil YD für Betriebsart Hand und Halt

YD für Betriebsart Hand und Halt wird wie folgt ermittelt:

$$YD = 0$$

Laufzeitfehler

Fehlermeldung

Es erfolgt eine Fehlermeldung, wenn:

- eine unzulässige Gleitkommazahl am Eingang YMAN anliegt
- ymax < ymin ist.

PID_PF: PID-Regler mit paralleler Struktur

38

Übersicht

Einleitung

Diese Kapitel beschreibt den Baustein PID_PF.

Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Kurzbeschreibung	342
Darstellung	343
Parametrierung des PID_PF-Reglers	345
Betriebsarten	347
Detaillierte Formeln	348
Laufzeitfehler	349

Kurzbeschreibung

Funktionsbeschreibung

Der Funktionsbaustein realisiert einen PID-Regler in reiner Parallel-Struktur.

Zwischen Führungsgröße SP und der Regelgröße PV wird die Regeldifferenz ERR gebildet. Die Regeldifferenz ERR bewirkt eine Änderung der Stellgröße Y.

Als zusätzliche Parameter können EN und ENO projektiert werden.

Eigenschaften

Der Funktionsbaustein besitzt folgenden Eigenschaften:

- PID-Regler in reiner Parallel-Struktur
- Unabhängige Verstärkungsfaktoren für P-. I- und D-Anteil
- Jeder Anteil P, I oder D kann einzeln abgeschaltet werden
- Stellgrenzen-Begrenzung im Automatik-Betrieb
- Antiwindup-Maßnahme nur bei aktivem I-Anteil
- Anti-Windup-Reset
- Betriebsarten Hand, Halt, Automatik
- stoßfreie Umschaltung Hand/Automatik
- D-Anteil umschaltbar auf Eingangsgröße PV bzw. Regeldifferenz ERR
- D-Anteil mit variabler Zeitverzögerung

Übertragungsfunktion

Die Übertragungsfunktion lautet:

$$G(s) = kp + \frac{ki}{s} + \frac{kd \times s}{s + \frac{1}{td_lag}}$$

YD
YI
YP

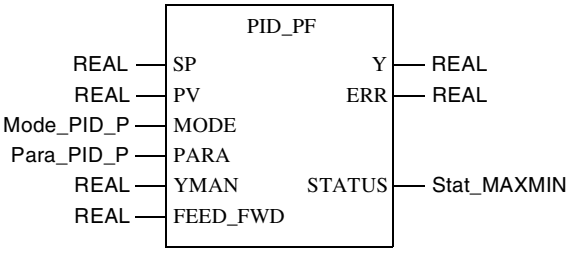
Erläuterung der Größen:

Größe	Bedeutung
YD	D-Anteil
YI	I-Anteil
YP	P-Anteil

Darstellung

Symbol

Darstellung des Bausteins:



Parameterbeschreibung PID_PF

Beschreibung der Bausteinparameter

Parameter	Datentyp	Bedeutung
SP	REAL	Führungsgröße
PV	REAL	Regelgröße
MODE	Mode_PID_P	Betriebsarten
PARA	Para_PID_P	Parameter
YMAN	REAL	Hand-Stellwert
FEED_FWD	REAL	Störgrößeneingang
Y	REAL	Stellgröße
ERR	REAL	Regeldifferenz
STATUS	Stat_MAXMIN	Status des Ausgangs Y

Parameterbeschreibung Mode_PID_P

Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
man	BOOL	"1": Betriebsart Hand
halt	BOOL	"1": Betriebsart Halt
d_on_pv	BOOL	"1": D-Anteil bezogen auf Regelgröße; "0": D-Anteil bezogen auf Regeldifferenz
reverse	BOOL	"1": Ausgang umgekehrt

Parameterbeschreibung Para_PID_P

Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
kp	REAL	Proportionalbeiwert (Verstärkungsfaktor = P-Anteil)
ki	REAL	Integrierbeiwert (Verstärkungsfaktor I-Anteil) [1/s]
kd	REAL	Differenzierbeiwert (Verstärkungsfaktor D-Anteil) [s]
td_lag	TIME	Verzögerungszeit, D-Anteil
ymax	REAL	Obere Stellgrenze
ymin	REAL	Untere Stellgrenze

Parameterbeschreibung Stat_MAXMIN

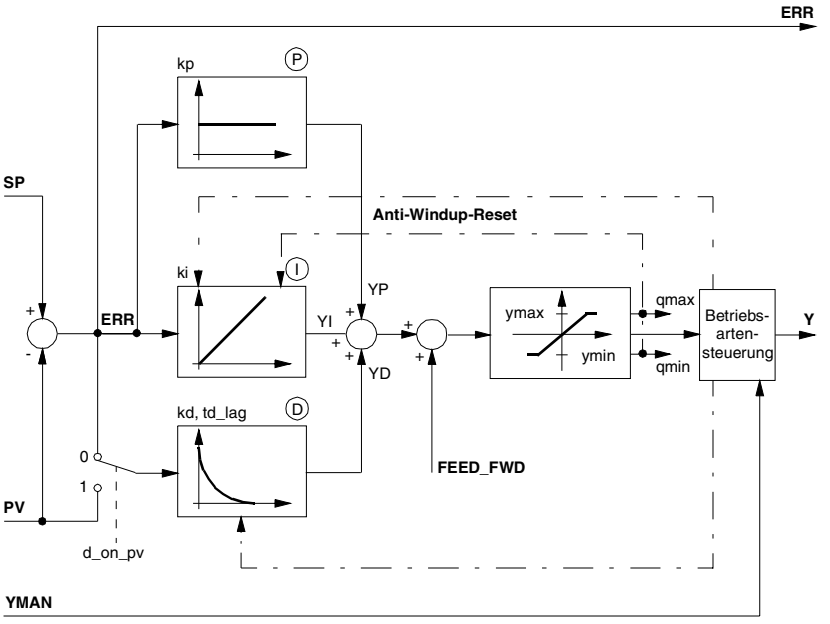
Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
qmax	BOOL	"1" = Y hat obere Stellgrenze erreicht
qmin	BOOL	"1" = Y hat untere Stellgrenze erreicht

Parametrierung des PID_PF-Reglers

Strukturbild

Nachfolgend finden Sie das Strukturbild des Bausteins PID_PF:



Parametrierung

Die Struktur des PID_PF-Reglers ist im *Strukturbild, S. 345* dargestellt.

Die Parametrierung des PID_PF-Reglers erfolgt zunächst durch die reinen PID-Parameter, nämlich den Proportionalbeiwert kp, den Integrierbeiwert ki und den Differenzierbeiwert kd.

Der P-, I- und D-Anteil können einzeln abgeschaltet werden, indem der entsprechende Eingang (kp, ki oder kd) auf 0 gesetzt wird.

Der D-Anteil wird mit der Zeitkonstante td_lag verzögert. Der D-Anteil kann entweder basierend auf der Regeldifferenz ERR (d_on_pv =0) oder basierend auf der Regelgröße PV (d_on_pv = 1) gebildet werden. Wird der D-Anteil aufgrund der Regelgröße PV bestimmt, so wird bei Führungsgrößenänderungen (Änderungen im Eingang SP) kein Sprung bedingt durch den D-Anteil entstehen. Der D-Anteil wirkt sich im Prinzip nur auf Störungen bzw. Prozeßänderungen aus.

Umkehr des Regelsinns

Ein entgegengesetztes Verhalten des Reglers kann durch Setzen des Eingangs reverse auf 1 erreicht werden. reverse = 0 bewirkt, daß bei einer positiven Störgröße der Ausgangswert steigt. reverse = 1 bewirkt, daß bei einer positiven Störgröße der Ausgangswert fällt.

Stellgrößen-Begrenzung

Die Grenzen ymax und ymin begrenzen den Ausgang nach oben sowie nach unten. Damit gilt ymin ≤ Y ≤ ymax.

Das Erreichen der Grenzwerte, bzw. eine Begrenzung des Ausgangssignals wird durch die Ausgänge qmax und qmin angezeigt:

- qmax = 1 wenn Y ≥ ymax
- qmin = 1 wenn Y ≤ ymin

Die Obergrenze ymax für die Begrenzung der Stellgröße ist größer als die Untergrenze ymin zu wählen, ansonsten meldet der Funktionsbaustein einen Fehler und arbeitet nicht.

Anti-Windup-Reset

Findet eine Stellgrößen-Begrenzung statt, soll durch Anti-Windup-Reset dafür gesorgt werden, daß der Integrale Anteil "nicht über alle Grenzen wachsen kann". Die Anti-Windup-Maßnahme wird nur dann durchgeführt, wenn der I-Anteil des Reglers nicht abgeschaltet ist. Die Grenzen für Anti-Windup sind hier dieselben, wie für die Stellgrößenbegrenzung. Der D-Anteil wird bei der Anti-Windup-Maßnahme nicht berücksichtigt, so daß Spitzen, hervorgerufen durch den D-Anteil, nicht durch die Anti-Windup-Maßnahme gekappt werden.

Die Anti-Windup-Maßnahme korrigiert den I-Anteil in der Form, daß gilt:

$$ymin - YP - FEED_FWD \leq YI \leq ymax - YP - FEED_FWD$$

Auswahl der Reglertypen

Es gibt verschiedene Reglertypen, die über die Parameter kp, ki und kd ausgewählt werden:

Reglertyp	kp	ki	kd
P-Regler	> 0	= 0	= 0
PI-Regler	> 0	> 0	= 0
PD-Regler	> 0	= 0	> 0
PID-Regler	> 0	> 0	> 0
I-Regler	= 0	> 0	= 0

Betriebsarten

Auswahl der Betriebsarten

Es gibt drei Betriebsarten, die über die Parameter man und halt ausgewählt werden:

Betriebsart	man	halt
Automatik	0	0
Hand	1	0 oder 1
Halt	0	1

Betriebsart Automatik

In der Betriebsart Automatik wird die Stellgröße Y durch den diskretisierten PID-Algorithmus, in Abhängigkeit von Regelgröße PV und Führungsgröße SP bestimmt. Die Stellgröße wird durch ymax und ymin begrenzt. Die Stellgrenzen sind gleichzeitig auch Grenzen für den Antiwindup-Reset.

Die Umschaltung von Automatik nach Hand ist normalerweise nicht stoßfrei, da der Ausgang Y jeden Wert zwischen ymax und ymin haben kann und Y bei der Umschaltung direkt den Wert YMAN annimmt.

Soll die Umschaltung von Automatik nach Hand trotzdem stoßfrei erfolgen, gibt es zwei Möglichkeiten, die für einen PID-Regler (siehe *Umschaltung von Automatik nach Hand, S. 312*) beispielhaft erklärt sind.

Betriebsart Hand

In der Betriebsart Hand wird der Handstellwert YMAN fest auf die Stellgröße Y übertragen. Die Stellgröße wird jedoch durch ymax und ymin begrenzt. Die internen Größen werden so nachgeführt, daß der Regler (bei zugeschaltetem I-Anteil) stoßfrei von Hand auf Automatik umgeschaltet werden kann. Die Stellgrenzen sind gleichzeitig auch Grenzen für den Antiwindup-Reset.

Der D-Anteil wird in dieser Betriebsart automatisch auf 0 gesetzt.

Betriebsart Halt

In der Betriebsart Halt bleibt der Reglerausgang, wie er vorgefunden wird, der Funktionsbaustein verändert die Stellgröße Y nicht (Regler bleibt stehen), d. h. Y = Y(old). Die internen Größen werden so nachgeführt, daß der Regler (bei zugeschaltetem I-Anteil) ohne Stoß von seiner aktuellen Lage aus weiterfährt. Die Stellgrenzen sind gleichzeitig auch Grenzen für den Antiwindup-Reset. Die Betriebsart Halt eignet sich unter anderem auch dafür, den Reglerausgang Y über ein externes Bediengerät zu stellen, wobei die internen Anteile im Regler richtig nachgeführt werden.

Der D-Anteil wird in dieser Betriebsart automatisch auf 0 gesetzt.

Detaillierte Formeln

Erläuterung der Formelgrößen

Bedeutung der Größen in den Formeln:

Größe	Bedeutung
dt	Zeitdifferenz zwischen aktullem Zyklus und vorherigem Zyklus
ERR	Regeldifferenz (SP - PV)
$ERR_{(new)}$	Wert der Regeldifferenz aus dem aktuellen Abtastschritt
$ERR_{(old)}$	Wert der Regeldifferenz aus dem vorherigen Abtastschritt
FEED_FWD	Störgröße
$PV_{(new)}$	Wert der Regelgröße aus dem aktuellen Abtastschritt
$PV_{(old)}$	Wert der Regelgröße aus dem vorherigen Abtastschritt.
Y	aktueller Ausgang (Betriebsart Halt) oder YMAN (Betriebsart Hand)
YD	D-Anteil
YI	I-Anteil
YP	P-Anteil

Stellgröße

Die Stellgröße setzt sich aus verschiedenen Teilgrößen zusammen:
$Y = YP + YI + YD + FEED_FWD$

Nach der Summation der Anteile findet eine Stellgrößen-Begrenzung statt, so daß gilt:

$ymin \leq Y \leq ymax$

Regeldifferenz

Die Regeldifferenz wird wie folgt ermittelt:

ERR = SP - PV, wenn reverse = 0

ERR = PV - SP, wenn reverse = 1

Übersicht zur Berechnung der Regler-Anteile

Nachfolgend finden Sie eine Übersicht über die unterschiedliche Berechnung der Regler-Anteile in Abhängigkeit von den Verstärkungsfaktoren kp, ki und kd:

- P-Anteil YP für Betriebsart Hand, Halt und Automatik
- I-Anteil YI für Betriebsart Automatik
- I-Anteil YI für Betriebsart Hand und Halt
- D-Anteil YD für Betriebsart Automatik
- D-Anteil YD für Betriebsart Hand und Halt

P-Anteil YP für alle Betriebsarten

YP für Betriebsart Hand, Halt und Automatik wird wie folgt ermittelt:

$$YP = kp \times ERR$$

I-Anteil YI für Betriebsart Automatik

YI für Betriebsart Automatik wird wie folgt ermittelt:

Für ki > 0 gilt:

$$YI_{(new)} = YI_{(old)} + ki \times dt \times \frac{ERR_{(new)} + ERR_{(old)}}{2}$$

Für ki = 0 gilt:

$$YI = 0$$

Der I-Anteil wird nach der Trapezregel gebildet.

I-Anteil YI für Betriebsart Hand und Halt

YI für Betriebsart Hand und Halt wird wie folgt ermittelt:

Für ki > 0 gilt:

$$YI = Y - (YP - FEED_FWD)$$

Für ki = 0 gilt:

$$YI = 0$$

D-Anteil YD für Betriebsart Automatik

YD für Betriebsart Automatik und Kaskade wird wie folgt ermittelt:

Für kd > 0 und d_on_pv = 0 gilt:

$$YD_{(new)} = \frac{td_lag}{dt + td_lag} \times (YD_{(old)} + kd \times (ERR_{(new)} - ERR_{(old)}))$$

Für kd > 0 und d_on_pv = 1 gilt:

$$YD_{(new)} = \frac{td_lag}{dt + td_lag} \times (YD_{(old)} + kd \times (PV_{(old)} - PV_{(new)}))$$

Für kd = 0 gilt:

$$YD = 0$$

D-Anteil YD für Betriebsart Hand und Halt

YD für Betriebsart Hand und Halt wird wie folgt ermittelt:

$$YD = 0$$

Laufzeitfehler

Fehlermeldung

Es erfolgt eine Fehlermeldung, wenn:

- eine unzulässige Gleitkommazahl am Eingang YMAN anliegt
- ymax < ymin ist.

PIDFF: Kompletter PID-Regler

39

Übersicht

Einleitung

Diese Kapitel beschreibt den Baustein PIDFF.

Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Kurzbeschreibung	352
Darstellung	353
Formeln	355
Strukturbild des PIDFF-Reglers	357
Parametrierung	358
Betriebsarten	363
Detaillierte Gleichungen	364
Detaillierte Gleichungen: Inkrementaler Algorithmus PID-Regler	367
Detaillierte Gleichungen: Inkrementaler Algorithmus Integral-Betrieb	369
Beispiele zum Baustein PIDFF	371
Laufzeitfehler	376

Kurzbeschreibung

Funktionsbeschreibung

Der Funktionsbaustein PIDFF basiert auf einem PID-Algorithmus mit paralleler oder gemischter Struktur (seriell / parallel).

Als zusätzliche Parameter können EN und ENO projektiert werden.

Funktionalitäten

Er hat zahlreiche Funktionalitäten aufzuweisen:

- Berechnung des Proportional-, Integral- und Differentialanteils in inkrementaler Form
- 2 Antiwindup-Maßnahmen
- Istwert, Sollwert und Ausgang in physikalischen Einheiten
- Aktion direkt oder invers
- Differentialanteil auf Istwert oder Abweichung
- Parametrierung der Übergangsverstärkung des Differentialanteils
- Gewichtung des Sollwertes beim Proportionalanteil (Dämpfung des Überlaufs)
- Möglichkeit, einen bausteinexternen Integralanteil aufzuschalten (RCPY-Eingang)
- Feed-Forward-Anteil zur Störungskompensation (FF-Eingang)
- Unempfindlichkeitszone auf Abweichung
- Inkremental- und Absolutwertausgang
- Obere und untere Begrenzung des Ausgangssignals (je nach Betriebsart)
- Gradientenbegrenzung des Ausgangssignals
- Ausgangs-Offset
- Auswahl der Betriebsart Hand / Automatik
- Betriebsart Tracking
- Obere und untere Begrenzung des Sollwertes

Ergänzende Funktionalitäten

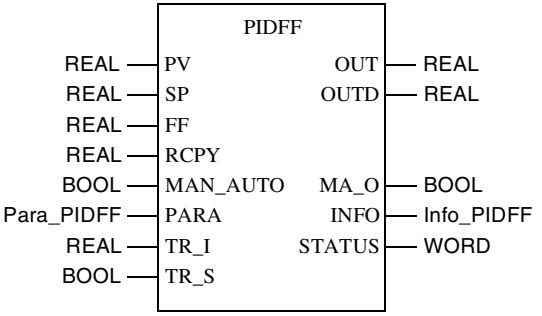
Andere Funktionsbausteine, die gemeinsam mit dem Baustein PIDFF zu verwenden sind, ergänzen diese Funktionalitäten:

- Automatische Reglereinstellung über den AUTOTUNE-Funktionsbaustein
- Auswahl eines internen oder externen Sollwertes über den Funktionsbaustein SP_SEL
- Steuerung des Handbetriebes der abgetasteten Regelschleifen (siehe *Abtasten, S. 36*) mit dem Funktionsbaustein MS

Darstellung

Symbol

Darstellung des Bausteins:



Parameterbeschreibung PIDFF

Beschreibung der Bausteinparameter

Parameter	Datentyp	Bedeutung
PV	REAL	Istwert (Process Value)
SP	REAL	Sollwert (Set Point)
FF	REAL	Störgrößeneingang
RCPY	REAL	Kopie der aktuellen Stellgröße
MAN_AUTO	BOOL	Regler-Betriebsart: "1": Automatik-Betrieb "0": Handbetrieb
PARA	Para_PIDFF	Parameter
TR_I	REAL	Initialisierungseingang
TR_S	BOOL	Initialisierungsbefehl
OUT	REAL	Absolutwertausgang
OUTD	REAL	Inkrementalwertausgang: Differenz zwischen dem Ausgang des aktuellen und des vorherigen Zyklus
MA_O	BOOL	Aktuelle Betriebsart des Funktionsbausteins: "1": Betriebsart Automatik "0": andere Betriebsart (d.h. Hand- oder Trackingbetrieb)
INFO	Info_PIDFF	Informationen
STATUS	WORD	Statuswort

Parameterbeschreibung Para_PIDFF

Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
id	UINT	Reserviert für automatische Reglereinstellung
pv_inf	REAL	Untere Grenze des Istwertbereichs
pv_sup	REAL	Obere Grenze des Istwertbereichs
out_inf	REAL	Untere Grenze des Ausgangswertbereichs
out_sup	REAL	Obere Grenze des Ausgangswertbereichs
rev_dir	BOOL	"0": direkte Aktion des PID-Reglers "1": inverse Aktion des PID-Reglers
mix_par	BOOL	"1": PID-Regler mit paralleler Struktur "0": PID-Regler mit gemischter Struktur
aw_type	BOOL	"1": Antiwindup-Halt wird gefiltert
en_rcpy	BOOL	"1": der RCPY-Eingang wird verwendet
kp	REAL	Proportionalbeiwert (Verstärkungsfaktor)
ti	TIME	Nachstellzeit
td	TIME	Vorhaltezeit
kd	REAL	Differenzierverstärkung
pv_dev	BOOL	Typ des Differentialanteils: "1": Differentialanteil bezogen auf Regeldifferenz "0": Differentialanteil bezogen auf Regelgröße (Istwert)
bump	BOOL	"1": Übergang auf Automatikbetrieb mit Stoß "0": Übergang auf Automatikbetrieb stoßfrei
dband	REAL	Unempfindlichkeitszone auf Abweichung
gain_kp	REAL	Dämpfung des Proportionalanteils innerhalb der Unempfindlichkeitszone dband
ovs_att	REAL	Dämpfung des Überlaufs
outbias	REAL	Manuelle Kompensation der statischen Abweichung
out_min	REAL	Unterer Grenzwert des Ausgangs
out_max	REAL	Oberer Grenzwert des Ausgangs
outrate	REAL	Grenzwert für Ausgangsänderung, in Einheiten pro Sekunde (≥ 0)
ff_inf	REAL	Unterer Grenzwert des FF-Bereichs
ff_sup	REAL	Oberer Grenzwert des FF-Bereichs
otff_inf	REAL	Unterer Grenzwert des out_ff Bereichs
otff_sup	REAL	Oberer Grenzwert des out_ff Bereichs

Parameterbeschreibung Info_PIDFF

Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
dev	REAL	Wert der Abweichung (PV - SP)
out_ff	REAL	Wert des Feed-Forward-Anteils

Formeln

Übertragungsfunktion

Je nachdem, ob die gemischte oder parallele Struktur verwendet wird, ist die Übertragungsfunktion folgende:

Struktur	Formeln
Gemischt	$$OUT = kp \times \left[1 + \frac{1}{ti \times p} + \frac{td \times p}{1 + \left(\frac{td}{kd}\right) \times p} \right] \times IN$$
Parallel	$$OUT = \left[kp + \alpha \times \frac{1}{ti \times p} + \alpha \times \frac{td \times p}{1 + \left(\frac{td}{kd}\right) \times p} \right] \times IN$$
mit α = Skalierungsfaktor	$$OUT = \frac{out_sup - out_inf}{pv_sup - pv_inf}$$

Berechnungsformeln

Die tatsächlich verwendeten Formeln variieren, je nachdem, ob der Funktionsbaustein die inkrementale oder absolute Form des Algorithmus verwendet.

Sehr vereinfacht kann der Funktionsbaustein eine der folgenden Formeln verwenden:

Algorithmus	ti	Formeln
Absolut	0	$OUT = TermP + TermD + TermFF + outbias$ $OUTD = OUT(new) - OUT(old)$
Inkremental	>0	$OUTD = TermP + TermI + TermD + TermFF$ $OUT = OUT(old) + OUTD(new)$

Erläuterung der Formelgrößen

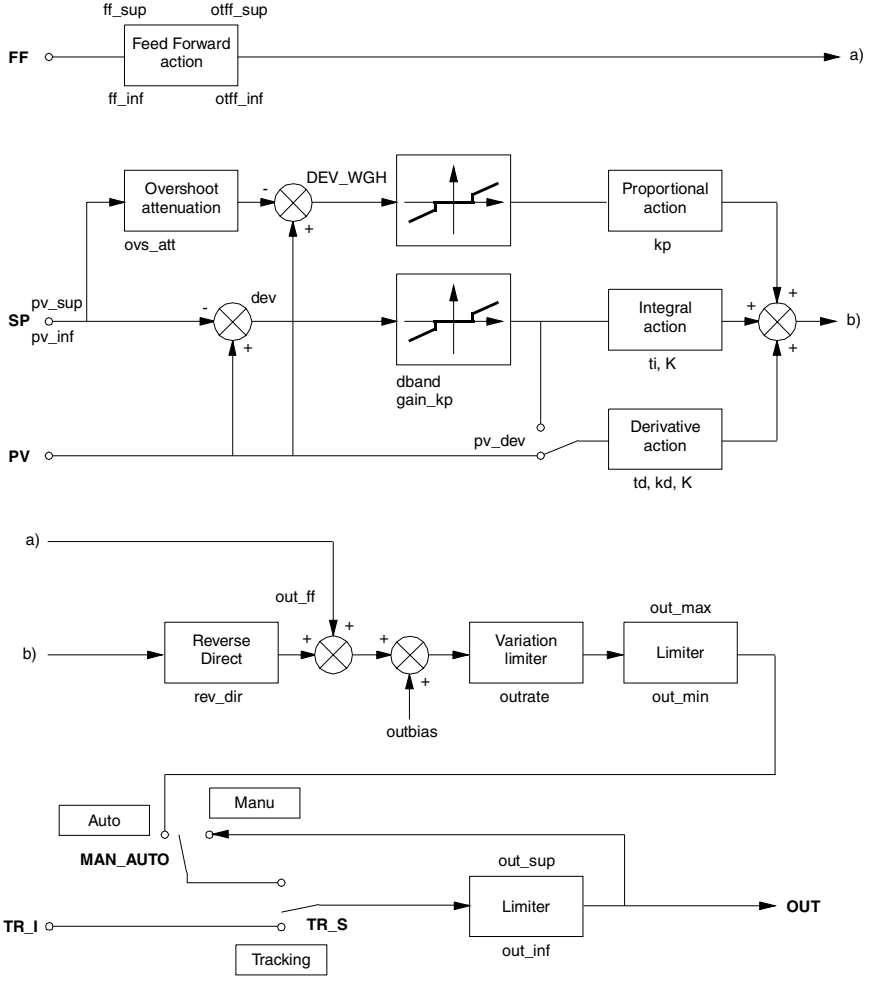
In der folgenden Tabelle finden Sie die Bedeutung der Formelgrößen:

Größe	Bedeutung
(new)	Wert, der bei der aktuellen Ausführung des Funktionsbausteins berechnet wird
(old)	Wert, der bei der vorherigen Ausführung des Funktionsbausteins berechnet wurde
OUT	Absolutwertausgang
OUTD	Inkrementalwertausgang
TermD	Wert des Differentialanteils
TermFF	Wert des Feed Forward-Anteils (Störungskompensation)
TermI	Wert des Integralanteils
TermP	Wert des Proportionalanteils

Strukturbild des PIDFF-Reglers

Strukturbild

Strukturbild des PIDFF-Reglers



Parametrierung

Gemischte/ parallele Struktur (mix_par)

Die Auswahl der Struktur erfolgt über den Parameter mix_par :

Wenn...	Dann ...
mix_par = 0	handelt es sich um eine gemischte Struktur, d.h. daß der Proportionalanteil im Anschluß an den Integral- und Differentialanteil aufgeschlagen wird. Der auf die Anteile aufgeschlagene Verstärkungsfaktor K (siehe *Strukturbild, S. 357*) entspricht kp.
mix_par = 1	ist die Struktur parallel, d.h. der Proportionalbeiwert wird parallel zum Integral- und Differentialbeiwert aufgeschlagen. In diesem Fall bezieht sich der Verstärkungsfaktor kp nicht auf den Integral- und den Differentialanteil. Die Verstärkung K entspricht in diesem Fall dem Verhältnis des Ausgabebereichs zum Meßbereich.

Absolute Algorithmen (ti = 0)

Absolute Algorithmen werden verwendet, wenn kein Integralanteil aufgeschlagen wird (ti = 0). In diesem Fall wird der Ausgang OUT zuerst berechnet und die Ausgangsänderung anschließend subtrahiert.

Inkrementale Algorithmen (ti > 0)

Inkrementale Algorithmen werden verwendet, wenn ein Integralanteil vorhanden ist (d.h. wenn ti > 0). Die Besonderheit dieses Algorithmus besteht darin, zunächst einmal die Ausgangsänderung OUTD zu berechnen und davon den Absolutwert-ausgang entsprechend folgender Formel zu errechnen:

$$OUT(new) = OUT(old) + OUTD$$

Diese Algorithmusform ermöglicht es, dem Regler einen SERVO-Funktionsbaustein nachzuschalten und so eine astatische Regelung zu erreichen.

Die inkrementale Form bietet darüber hinaus folgende Möglichkeiten:

Möglichkeit	**Erläuterung**
Bausteinexterner Integralanteil (mit en_rcpy = 1)	Wenn der reale Anteil vom durch den Regler berechneten Wert abweicht (bei offenem Regelkreis), sollte bei der Berechnung der reale Wert zugrunde gelegt werden. Wenn dieser Wert verfügbar ist, ist er auf den RCPY-Eingang aufzuschalten und der Parameter en_rcpy auf 1 zu setzen. Bei den vom Funktionsbaustein durchgeführten Berechnungen wird die Gleichung OUT(new) = OUT (old) + OUTD zu OUT(new) = RCPY + OUTD Diese Einrichtung ist insbesondere bei Kaskaden- oder kaskadenähnlicher Regelung sinnvoll. **Hinweis**: In diesem Fall wird der Ausgang OUT nicht begrenzt.
Erweiterte Antiwindup-Maßnahme	Die inkrementale Form des PID-Reglers bietet standardmäßig eine im Algorithmus berücksichtigte Antiwindup-Maßnahme. Bei aw_type = 0 wird dieser Typ zugrunde gelegt. In diesem Fall kann es vorkommen, daß der Ausgang gesättigt ist und seinen Grenzwert plötzlich verläßt, auch wenn das Vorzeichen der Abweichung sich nicht ändert (z.B. unter dem Einfluß einer kurzen Störung während der Messung). Es ist möglich, eine zweite Antiwindup-Maßnahme einzusetzen (aw_type = 1), die verhindert, daß der Ausgang seinen Grenzwert übersteigt, solange die Abweichung das Vorzeichen nicht ändert.

Gewichtung des Sollwertes beim Proportionalanteil (Dämpfung des Überlaufs)

Wenn ein Integralanteil vorhanden ist (ti > 0), ermöglicht der Parameter ovs_att die Gewichtung des Proportionalanteils: die Berechnung des Proportionalanteils basiert auf der gewichteten Abweichung ($PV - (1 - ovs_att) \times SP$).

Dies ermöglicht eine Einflußnahme im Falle eines Überlaufs, wie er bei Sollwertänderungen auftreten kann. Ziel ist es, einen regelintensiven Proportionalanteil und damit eine dynamische Antwort auf Störungen zu erhalten, ohne daß es bei der Regelung zu Überläufen kommt.

Der Parameter ovs_att kann kontinuierlich schwanken zwischen:

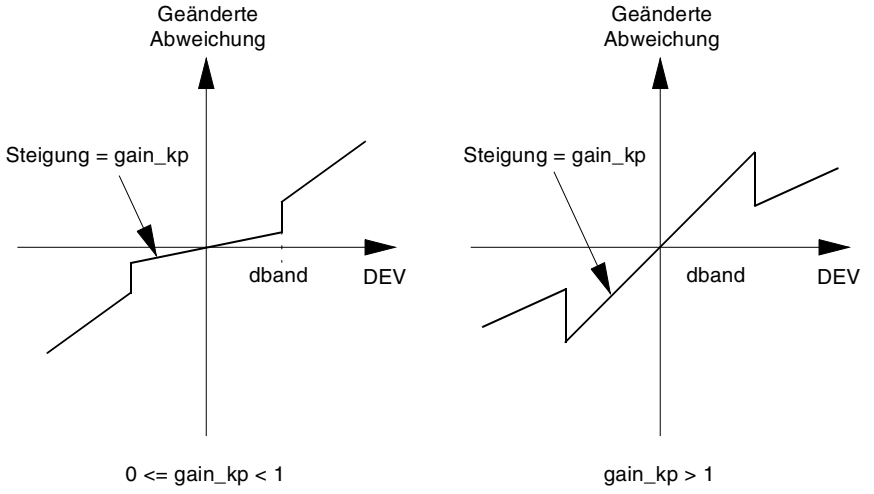
Wert	Bedeutung
0	der Abweichung (Regeldifferenz) aufgeschalteter Proportionalanteil (klassischer Fall)
1	dem Meßwert (Regelgröße) aufgeschalteter Proportionalanteil (bei empfindlichen Prozessen oder integral wirkenden Prozessen).

Unempfindlichkeitszone auf Abweichung (dband)

Bei Erreichen des Arbeitspunktes ermöglicht es die Unempfindlichkeitszone, kleinere Angleichstöße an den Wert des Stellgliedes zu begrenzen: solange die Abweichung betragsmäßig dband unterschreitet, legt der Funktionsbaustein bei der Berechnung den Wert Null zugrunde.

Innerhalb der Unempfindlichkeitszone kann der erweiterte Parameter gain_kp zur Änderung der Abweichung verwendet werden. Dies ist besser, als diese zu löschen. Die geänderte Abweichung (multipliziert mit gain_kp) wird zur Berechnung des Proportional- und des Integralanteils verwendet.

Darstellung der Veränderung der Abweichung



0 <= gain_kp < 1

gain_kp > 1

Übergangsverstärkung beim Differentialanteil

Der Funktionsbaustein PIDFF beinhaltet einen Filter erster Ordnung für den Differentialanteil. Die Filterverstärkung kd ist parametrierbar, wodurch sowohl solche Prozesse verarbeitet werden können, wo der Differentialanteil sehr stark gefiltert werden muß, wie auch Prozesse, wo die Filterung des Differentialanteils aufgehoben werden kann, da das Signal "rein" genug ist.

Feed-Forward-Anteil zur Störungs-kompensation (FF-Eingang)

Bei einer klassischen PID-Regelung reagiert der Regler auf Ausgangsänderungen des Regelprozesses (geschlossener Regelkreis). Im Störungsfall reagiert der Regler nur, wenn der Istwert vom Sollwert abweicht. Die Feed-Forward-Funktion ermöglicht die Kompensation einer meßbaren Störung sofort bei Auftreten. Diese als offener Regelkreis konzipierte Funktion beseitigt die Auswirkung der Störung: in diesem Falle spricht man von einer Störgrößenaufschaltung (Feed Forward).

Der Anteil des Feed-Forward-Eingangs an der Stellgröße des Reglers wird nach Einbeziehung des Reglersinns direkt/invers aufgeschlagen.

Die Berechnung erfolgt nach folgender Formel:

$$\text{out_ff} = \frac{(\text{FF} - \text{ff_inf}) \times (\text{otff_sup} - \text{otff_inf})}{(\text{ff_sup} - \text{ff_inf})} + \text{otff_inf}$$

Ein konkretes Anwendungsbeispiel dieser Funktion ist im Abschnitt "*Anwendungsbeispiel der Feed Forward-Funktion, S. 371*" dargestellt.

Hinweis: Wenn ff_sup = ff_inf, wird die Berechnung des Feed_Forward-Anteils ignoriert.

Weitere Eigenschaften

Der Baustein besitzt folgende weitere Eigenschaften:

- Der Parameter outbias ermöglicht die Präzisierung am Arbeitspunkt, wenn der Prozeß keinen Integralanteill beinhaltet (ti = 0).
- Im Automatikbetrieb ist der Ausgang OUT auf den Bereich zwischen out_min und out_max begrenzt, im Hand- und Trackingbetrieb auf den Bereich zwischen out_inf und out_sup. Wenn der vom Funktionsbaustein berechnete Wert (oder vom Anwender im Handbetrieb geschriebene Wert) einen dieser Grenzwerte überschreitet, wird der Wert von OUT gekappt. Der Inkrementalwertausgang OUT_D berücksichtigt diese Kappung jedoch nie. Dies ermöglicht es dem Funktionsbaustein PIDFF, einen SERVO-Funktionsbaustein zu steuern, ohne daß die Stellgliedposition rückgeführt werden muß (kontinuierliche Regelung).
- Die Anstiegsgeschwindigkeit des Ausgangs wird durch den Parameter outrate begrenzt.
- Die Wahlmöglichkeit direkte/inverse Aktion (Parameter rev_dir) ermöglicht es, den Reglersinn der Verbindung Stellglied/Prozeß anzupassen.
- Der Differentialanteil kann sich sowohl auf den Istwert (pv_dev = 0), wie auch auf die Abweichung (pv_dev = 1) auswirken.
- pv_inf und pv_sup entsprechen dem oberen und unteren Grenzwert des Sollwertes.
- Der Funktionsbaustein kann auch im reinen Integralbetrieb wirken (mit kp = 0).

Betriebsarten

Auswahl der Betriebsarten

Der Funktionsbaustein PIDFF verfügt über 3 Betriebsarten: Automatik, Hand und Tracking. Wie nachstehende Tabelle zeigt, hat der Tracking-Betrieb Vorrang vor den anderen Betriebsarten.

Die Betriebsarten werden über die Eingänge MAN_AUTO und TR_S ausgewählt:

Betriebsart	TR_S	MAN_AUTO	Bedeutung
Automatik	0	1	Die Ausgänge OUT und OUTD entsprechen dem Ergebnis der vom Funktionsbaustein durchgeführten Berechnungen. Die Grenzwerte für den Ausgang OUT sind out_min und out_max.
Hand	0	0	Der Ausgang OUT wird nicht über den Funktionsbaustein gesetzt. Der Anwender kann dessen Wert direkt ändern. OUT bleibt jedoch begrenzt, in dieser Betriebsart handelt es sich dabei um die Grenzwerte out_inf und out_sup (statt out_min und out_max im Automatikbetrieb).
Tracking	1	0 oder 1	Der Eingang TR_I wird auf den Ausgang OUT geführt. Wie im Handbetrieb liegt OUT zwischen den Grenzwerten out_inf und out_sup.

Umschaltung Hand -> Automatik oder Tracking -> Automatik

Die Art der Umschaltung ist abhängig von bump:

Wenn...	Dann ...
bump = 0	erfolgt die Umschaltung stoßfrei. **Hinweis:** Bei ti = 0 wird der Parameter outbias neu berechnet. So können die OUT-Werte stoßfrei beginnend beim letzten Wert der vorherigen Betriebsart weiterstarten.
bump = 1	kann die Umschaltung mit Stoß erfolgen.

Detaillierte Gleichungen

Übersicht

In diesem Abschnitt werden zu folgenden Fällen die detaillierten Gleichungen dargestellt:

- Konvention der wichtigsten in den Gleichungen verwendeten Zwischenvariablen und Funktionen
- *Absoluter Algorithmus, S. 366*
- *Inkrementaler Algorithmus PID-Regler, S. 367*
  - Normale inkrementale Algorithmen (aw_type = 0)
  - Mit stoßfreier AntiwindupMaßnahme (aw_type = 1)
- *Inkrementaler Algorithmus Integral-Betrieb, S. 369*
  - Normale inkrementale Algorithmen (aw_type = 0)
  - Mit stoßfreier Antiwindup-Maßnahme (aw_type = 1)

Konvention

In den folgenden Gleichungen werden verschiedene Variablen und Funktionen verwendet. Die den Parametern des Funktionsbausteins entsprechenden Variablen werden nicht erneut beschrieben.

Die wichtigsten Zwischenvariablen und die verwendeten Funktionen werden in den folgenden Tabellen beschrieben.

Erläuterung der Zwischen-variablen

Hier finden Sie eine Erläuterung der wichtigsten Zwischenvariablen:

Zwischenvariable	Bedeutung
DEV_WGH	DEV_WGH = PV - (1 - ovs_att) * SP
dt	Zeit, die seit der letzten Ausführung des Funktionsbausteins abgelaufen ist
K	Verstärkungsfaktor des Integral- und Differentialanteils. Die Verstärkung variiert je nach Struktur des Funktionsbausteins (gemischt oder parallel) und hängt davon ab, ob der Proportionalanteil aufgeschaltet wird oder nicht: ● Wenn mix_par = 0 (gemischte Struktur) und kp <> 0, gilt K = kp ● Wenn mix_par = 1 (parallele Struktur) oder kp = 0, gilt:) K = Skalierfaktor = $\alpha = \frac{out_sup - out_inf}{pv_sup - pv_inf}$
(new)	Wert, der bei der aktuellen Ausführung des Funktionsbausteins berechnet wird
(old)	Wert, der bei der vorherigen Ausführung des Funktionsbausteins berechnet wurde
OUTc	Vor Begrenzung berechneter Ausgangswert
sense	Reglersinn
TermAW	Wert der stoßfreien Antiwindup-Maßnahme
TermD	Wert des Differentialanteils
TermFF	Wert des Feed-Forward-Anteils (Störungskompensation)
TermI	Wert des Integralanteils
TermP	Wert des Proportionalanteils
VAR	Zur Berechnung des Differentialanteils verwendete Variable. Ihr Wert ist abhängig vom Parameter pv_dev : ● Wenn pv_dev = 0, VAR = PV ● Wenn pv_dev = 1, VAR = dev

Erläuterung der Funktionen

Hier finden Sie eine Erläuterung der wichtigsten Funktionen:

Funktion	Bedeutung
Reglersinn	Der Reglersinn hat folgende Wirkrichtungen: • +1 Es handelt sich um eine direkte Aktion (rev_dir = 0), d.h. daß eine positive Abweichung (PV - SP) ein Anwachsen des Ausgangswertes generiert. • -1 Es handelt sich um eine inverse Aktion (rev_dir = 1), d.h. daß eine positive Abweichung (PV - SP) eine Verringerung des Ausgangswertes generiert.
Funktion Δ	$\Delta(x(t)) = x(t) - x(t-1)$
'Begrenzen'	Begrenzungsfunktion für den Ausgang des Funktionsbausteins

Absoluter Algorithmus

Folgende Gleichungen gelten mit PD-Reglern (ti = 0);

$$OUT = TermP + TermD + TermFF + outbias$$

$$OUTD = OUTP(new) - OUTP(old)$$

$$OUT = limiter(OUT)$$

Wert des Proportionalanteils TermP

$$TermP = sense \times kp \times dev$$

Wert des Differentialanteils TermD

$$TermD = sense \times \frac{td \times TermD_{(old)} + K \times td \times kd \times (VAR_{(new)} - VAR_{(old)})}{kd \times dt + td}$$

Wert des Feed-Forward-Anteils TermFF

$$TermFF = \frac{(FF - ff_inf) \times (otff_sup - otff_inf)}{ff_sup - ff_inf} + otff_inf$$

Detaillierte Gleichungen: Inkrementaler Algorithmus PID-Regler

Inkrementaler Algorithmus PID-Regler

Für PID-Regler (ti > 0) werden die Gleichungen in Abhängigkeit des Elements aw_type nach folgenden Fällen unterschieden:

Element	Bedeutung
aw_type = 0	Normale inkrementale Algorithmen
aw_type = 1	Mit stoßfreier Antiwindup-Maßnahme

PID-Regler: aw_type = 0

Folgende Gleichungen gelten für normale inkrementale Algorithmen von PID-Reglern;

$$OUTD = TermP + TermI + TermD + TermFF$$

$$OUT = limiter(OUT)$$

Wenn en_rcpy = 0 ist, gilt:

$$OUT = OUT(old) + OUTD(new)$$

Wenn en_rcpy = 1 ist, gilt:

$$OUT = RCPY + OUTD(new)$$

Wert des Proportionalanteils TermP:

$$TermP = sense \times kp \times [\Delta(DEV_WGH)]$$

Wert des Integralanteils TermI:

$$TermI = sense \times kp \times \frac{dt}{ti} \times dev$$

Wert des Differentialanteils TermD

$$TermD = \Delta\left[sense \times \frac{td \times TermD_{(old)} + K \times td \times kd \times (VAR_{(new)} - VAR_{(old)})}{kd \times dt + td}\right]$$

Wert des Feed-Forward-Anteils TermFF

$$TermFF = \Delta\left[\frac{(FF - ff_inf) \times (otff_sup - otff_inf)}{(ff_sup - ff_inf)} + otff_inf\right]$$

PID-Regler: aw_type = 1

Folgende Gleichungen gelten für inkrementale Algorithmen von PID-Reglern mit stoßfreier Antiwindup-Maßnahme;

$$OUTD = TermP + TermI + TermD + TermFF + TermAW$$

$$OUTc = OUTc(old) + OUTD(new)$$

$$OUT = limiter(OUTc)$$

Wert des Proportionalanteils TermP:

$$TermP = sense \times kp \times [\Delta(DEV_WGH)]$$

Wert des Integralanteils TermI:

$$TermI = sense \times kp \times \frac{dt}{ti} \times dev$$

Wert des Differentialanteils TermD

$$TermD = \Delta\left[sense \times \frac{td \times TermD_{(old)} + K \times td \times kd \times (VAR_{(new)} - VAR_{(old)})}{kd \times dt + td}\right]$$

Wert des Feed-Forward-Anteils TermFF

$$TermFF = \Delta\left[\frac{(FF - ff_inf) \times (otff_sup - otff_inf)}{(ff_sup - ff_inf)} + otff_inf\right]$$

Wert der stoßfreien Antiwindup-Maßnahme TermAW

Wenn en_rcpy = 0 ist, gilt:

$$TermAW = \frac{dt}{ti}[OUT(old) - OUTc(old)]$$

Wenn en_rcpy = 1 ist, gilt:

$$TermAW = \frac{dt}{ti}[RCPY - OUTc(old)]$$

Detaillierte Gleichungen: Inkrementaler Algorithmus Integral-Betrieb

Inkrementaler Algorithmus Integral-Betrieb

Der Regler kann im reinen Integralbetrieb eingesetzt werden (kp = 0).

Auch hier werden die Gleichungen in Abhängigkeit des Elements aw_type nach folgenden Fällen unterschieden:

Element	Bedeutung
aw_type = 0	Normale inkrementale Algorithmen
aw_type = 1	Mit stoßfreier Antiwindup-Maßnahme

Integral-Betrieb: aw_type = 0

Folgende Gleichungen gelten für normale inkrementale Algorithmen von Reglern im Integral-Betrieb;

$$OUTD = TermI + TermFF$$

$$OUT = limiter(OUT)$$

Wenn en_rcpy = 0 ist, gilt:

$$OUT = OUT(old) + OUTD(new)$$

Wenn en_rcpy = 1 ist, gilt:

$$OUT = RCPY + OUTD(new)$$

Wert des Integralanteils TermI:

$$TermI = sense \times \alpha \times \frac{dt}{ti} \times dev$$

Wert des Feed-Forward-Anteils TermFF

$$TermFF = \Delta\left[\frac{(FF - ff_inf) \times (otff_sup - otff_inf)}{(ff_sup - ff_inf)} + otff_inf\right]$$

Integral-Betrieb: aw_type = 1

Folgende Gleichungen gelten für inkrementale Algorithmen von Integral-Reglern mit stoßfreier Antiwindup-Maßnahme;

$$OUTD = TermI + TermFF + TermAW$$

$$OUTc = OUTc(old) + OUTD(new)$$

$$OUT = limiter(OUTc)$$

Wert des Integralanteils TermI:

$$TermI = sense \times \alpha \times \frac{dt}{ti} \times dev$$

Wert des Feed-Forward-Anteils TermFF

$$TermFF = \Delta\left[\frac{(FF - ff_inf) \times (otff_sup - otff_inf)}{(ff_sup - ff_inf)} + otff_inf\right]$$

Wert der stoßfreien Antiwindup-Maßnahme TermAW

Wenn en_rcpy = 0 ist, gilt:

$$TermAW = \frac{dt}{ti}[OUT(old) - OUTc(old)]$$

Wenn en_rcpy = 1 ist, gilt:

$$TermAW = \frac{dt}{ti}[RCPY - OUTc(old)]$$

Beispiele zum Baustein PIDFF

Beispiel-Übersicht

In diesem Kapitel finden Sie die folgenden Beispiele:

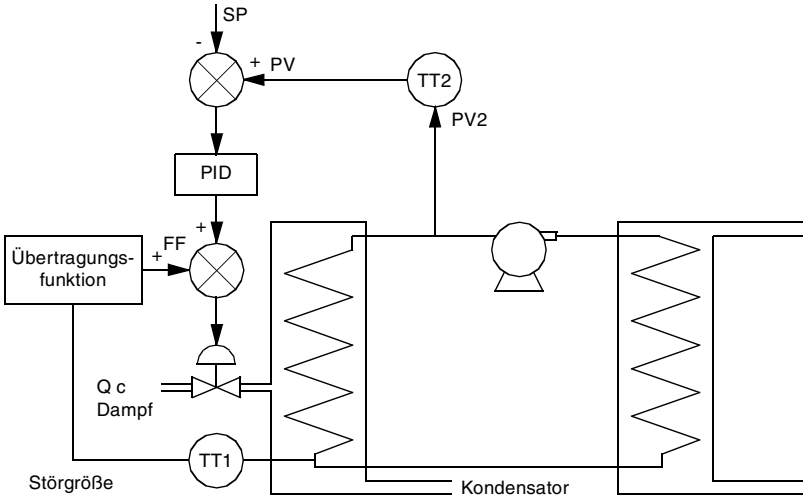
- *Anwendungsbeispiel der Feed Forward-Funktion, S. 371*
- Klassische Regelbeispiele, programmiert über den PIDFF-Funktionsbaustein:
  - *Beispiel zur kaskadierten Anordnung zweier Regler, S. 373*
  - *Beispiel einer kaskadenähnlichen Regelung, S. 375*

Anwendungsbeispiel der Feed Forward-Funktion

Bei einem Wärmetauscher soll am Ausgang des Sekundärkreislaufs die Temperatur PV2 geregelt werden. Ein PID-Regler steuert das Einlaufventil für Warmluft in Abhängigkeit von PV2 und dem Sollwert SP. Die Kaltwassertemperatur wird bei diesem Regelprozeß wie eine meßbare Störgröße betrachtet.

Die Feed-Forward-Funktion ermöglicht eine Reaktion, sobald die Kaltwassertemperatur sich ändert und nicht erst, wenn PV2 abgesunken ist.

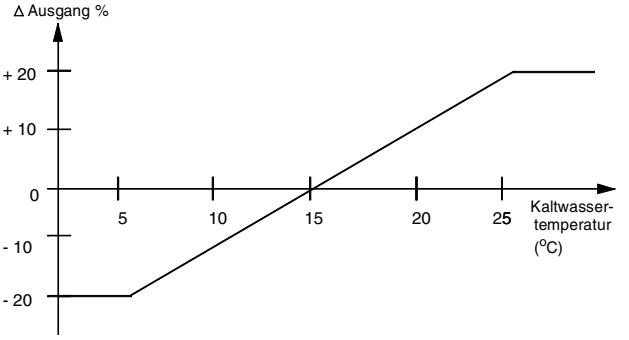
Darstellung des Regelkreises:



Folgende Hypothesen werden angenommen:

- Die Kondensatorausgangstemperatur (Kaltwasser-Temperatur) variiert zwischen 5 °C und 25 °C, mit einem Mittelwert von 15 °C.
- Eine Temperaturänderung DT wirkt sich vollständig auf die Ausgangstemperatur des Wärmetauschers aus.
- Um einen Temperaturanstieg (oder -abstieg) um 5 °C am Ausgang des Wärmetauschers zu kompensieren, muß das Dampfsteuerventil um 10 % geschlossen (oder geöffnet) werden.

Die Feed-Forward-Eingangsparameter sind so einzustellen, daß die Kaltwassertemperatur auf das Dampfsteuerventil folgenden Einfluß hat:

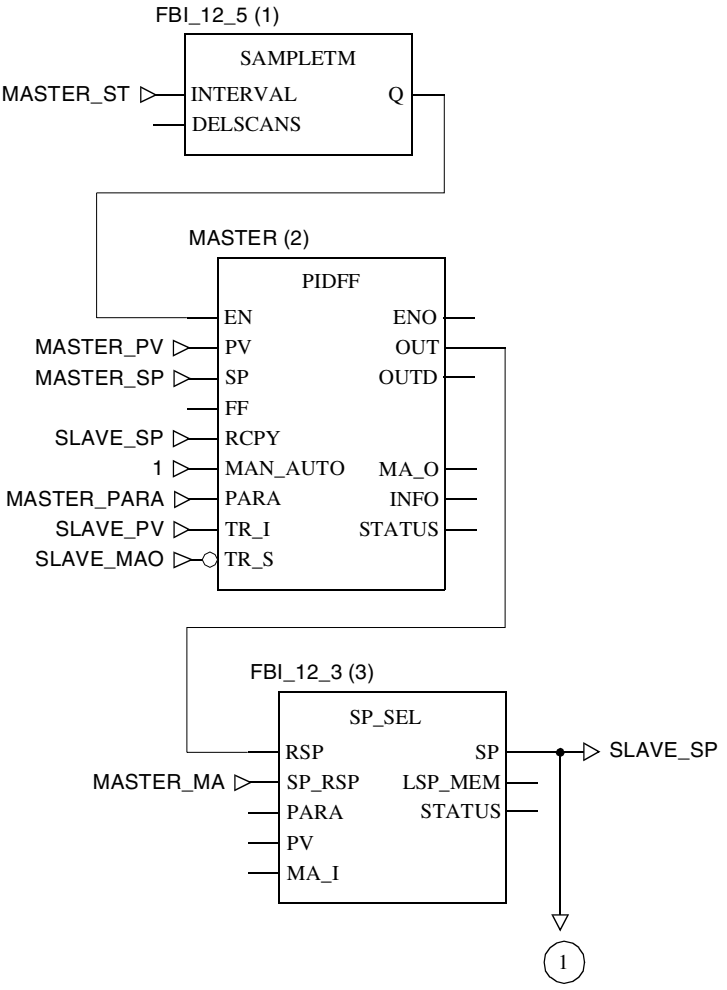
Temperaturbereich	**Auswirkungen**
15 °C	kein Einfluß
10% pro 5 °C zwischen 5 und 25 °C	

Vorzunehmende Einstellungen

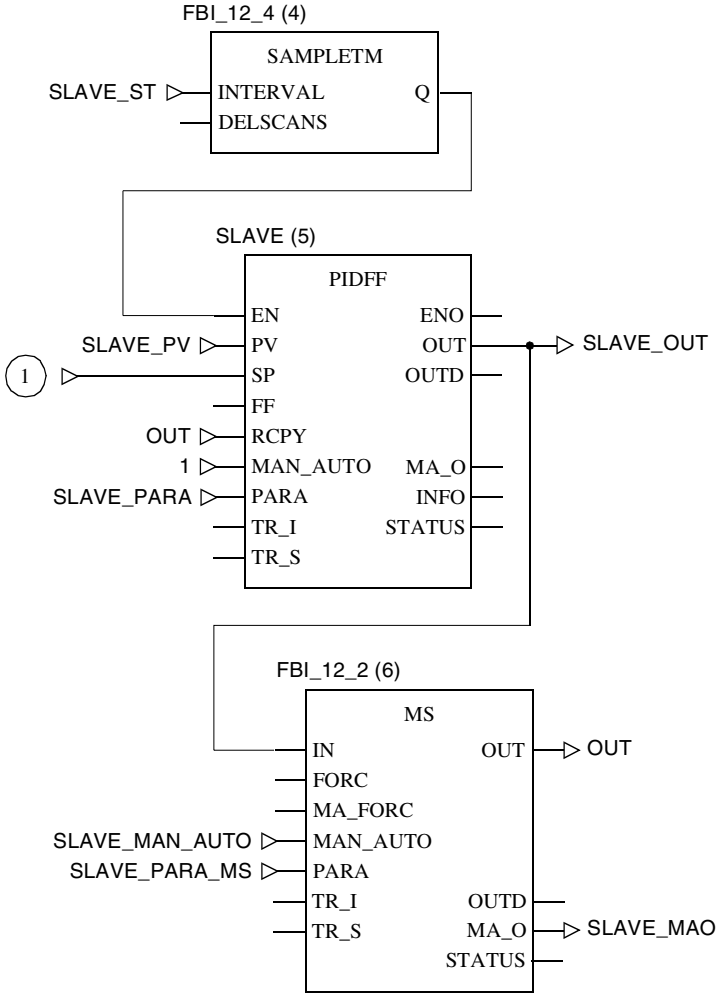
Element	**Wert**
ff_sup	25 °C
ff_inf	5 °C
otff_sup	10 %
otff_inf	- 10 %

Beispiel zur kaskadierten Anordnung zweier Regler

Nachfolgend finden Sie die Darstellung des Funktionsplans, Teil 1:

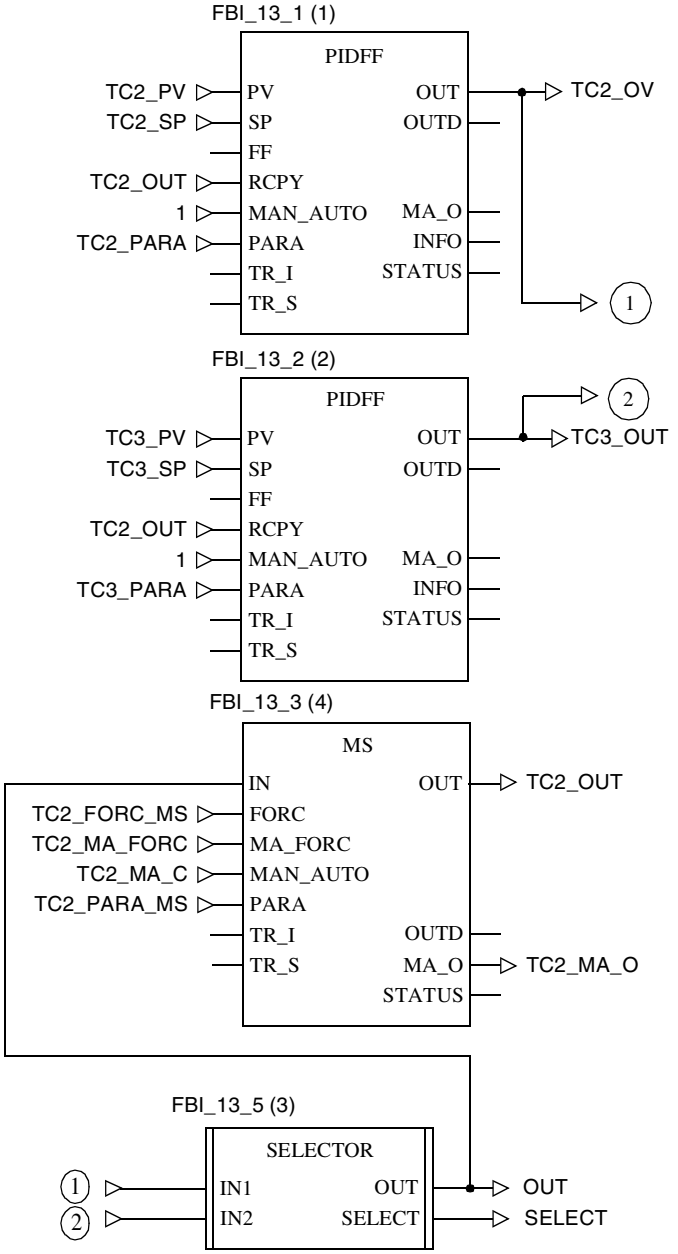


Nachfolgend finden Sie die Darstellung des Funktionsplans, Teil 2:



Beispiel einer kaskadenähnlichen Regelung

Nachfolgend finden Sie die Darstellung des Funktionsplans:



Laufzeitfehler

Statuswort

Im Statuswort werden folgenden Meldungen angezeigt:

Bit	Bedeutung
Bit 0 = 1	Fehler in einer Berechnung in Gleitkommawerten
Bit 1 = 1	Erfassung eines nicht zulässigen Wertes an einem Gleitkommawerteingang
Bit 2 = 1	Division durch Null bei einer Berechnung in Gleitkommawerten
Bit 3 = 1	Kapazitätsüberlauf bei einer Berechnung in Gleitkommawerten
Bit 4 = 1	Folgende Verhalten werden angezeigt: ● Der Eingang SP liegt außerhalb des Bereichs [pv_inf, pv_sup] : zur Berechnung verwendet der Funktionsbaustein den Wert pv_inf bzw. pv_sup. ● Einer der Parameter kp, dband, gain_kp, outrate ist negativ: der Funktionsbaustein verwendet den Wert 0 anstelle des fehlerhaften Parameterwertes. ● kd < 1 (mit td <> 0) : der Funktionsbaustein verwendet den Wert 1 anstelle des fehlerhaften Wertes von kd. ● Der Parameter ovs_att liegt außerhalb des Bereichs [0, 1]: zur Berechnung verwendet der Funktionsbaustein den Wert 0 bzw. 1. ● Einer der Parameter out_min oder out_max liegt außerhalb des Bereichs [out_inf, out_sup]. Der Funktionsbaustein verwendet zur Berechnung den Wert out_inf bzw. out_sup. ● Einer der Parameter outbias, otff_inf oder otff_sup liegt außerhalb des Bereichs [(out_min - out_max), (out_max - out_min)]. Der Funktionsbaustein verwendet zur Berechnung den Wert (out_min - out_max) bzw. (out_max - out_min).
Bit 5 = 1	Der Ausgang OUT hat den unteren Grenzwert out_min erreicht (siehe Hinweis)
Bit 6 = 1	Der Ausgang OUT hat den oberen Grenzwert out_max erreicht (siehe Hinweis)
Bit 7 = 1	Die Grenzwerte pv_inf und pv_sup sind identisch

Hinweis zum Ausgang OUT

Hinweis: Im Handbetrieb bleiben diese Bits nur einen Programmzyklus lang auf 1. Wenn der Anwender für OUT einen Wert eingibt, der einen der Grenzwerte überschreitet, setzt der Funktionsbaustein das Bit 5 oder 6 auf 1 und kappt den vom Bediener eingegebenen Wert. Bei der folgenden Ausführung des Funktionsbausteins liegt der Wert von OUT dann nicht mehr außerhalb des Bereichs und die Bits 5 bzw. 6 werden wieder zu Null gesetzt.

Fehlermeldung

Dieser Fehler wird angezeigt, wenn an einem Eingang ein nicht gleitender Wert festgestellt wird, bei einer Berechnung mit Gleitkommawerten ein Problem auftritt oder die Grenzwerte pv_inf und pv_sup des Reglers identisch sind. In diesem Fall bleiben die Ausgänge OUT, OUTD, MA_O und INFO unverändert.

Warnung

In folgenden Fällen wird eine Warnung ausgegeben:

- Einer der Parameter kp, dband, gain_kp, outrate ist negativ: Der Funktionsbaustein verwendet dann den Wert 0 anstelle des fehlerhaften Parameterwertes.
- kd < 1 (mit td <> 0): der Funktionsbaustein verwendet den Wert 1 anstelle des fehlerhaften Wertes von kd.
- Der Parameter ovs_att liegt außerhalb des Bereichs [0, 1]: der Funktionsbaustein verwendet zur Berechnung den Wert 0 bzw. 1.
- Der Parameter out_min bzw. out_max liegt außerhalb des Bereichs [out_inf, out_sup]. Der Funktionsbaustein verwendet zur Berechnungen den Wert out_inf bzw. out_sup.
- Einer der Parameter outbias, otff_inf bzw. otff_sup liegt außerhalb des Bereichs [(out_min - out_max), (out_max - out_min)]. Der Funktionsbaustein verwendet zur Berechnung den Wert (out_min - out_max) bzw. (out_max - out_min).

PIDP1: PID-Regler mit paralleler Struktur

40

Übersicht

Einleitung

Diese Kapitel beschreibt den Baustein PIDP1.

Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Kurzbeschreibung	380
Darstellung	381
Parametrierung des PIDP1-Reglers	383
Betriebsarten	385
Detaillierte Formeln	386
Laufzeitfehler	388

Kurzbeschreibung

Funktionsbeschreibung

Der Funktionsbaustein realisiert einen PID-Regler in reiner Parallel-Struktur.

Zwischen Sollwert SP und der Regelgröße PV wird die Regeldifferenz ERR gebildet. Die Regeldifferenz ERR bewirkt eine Änderung der Stellgröße Y.

Als zusätzliche Parameter können EN und ENO projektiert werden.

Eigenschaften

Der Funktionsbaustein besitzt folgenden Eigenschaften:

- PID-Regler in reiner Parallel-Struktur
- Jeder Anteil P, I oder D kann einzeln abgeschaltet werden
- Stellgrenzen-Begrenzung im Automatik-Betrieb
- Antiwindup-Maßnahme nur bei aktivem I-Anteil
- Anti-Windup-Reset
- Betriebsarten Hand, Halt, Automatik
- stoßfreie Umschaltung Hand/Automatik
- D-Anteil umschaltbar auf Eingangsgröße PV bzw. Regeldifferenz ERR
- D-Anteil mit variabler Zeitverzögerung

Übertragungsfunktion

Die Übertragungsfunktion lautet:

$$G(s) = KP + \frac{KI}{s} + \frac{KD \times s}{s + \frac{1}{TD_LAG}}$$

YD
YI
YP

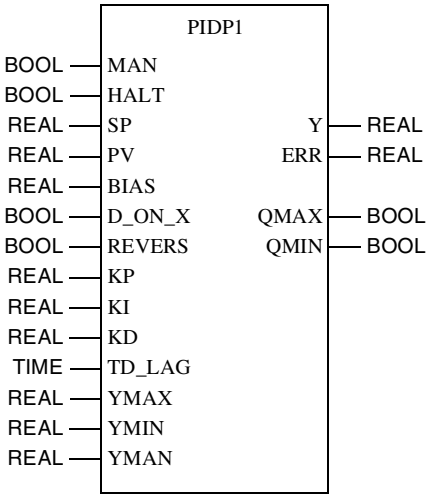
Erläuterung der Größen:

Größe	Bedeutung
YD	D-Anteil
YI	I-Anteil
YP	P-Anteil

Darstellung

Symbol

Darstellung des Bausteins:



Parameterbeschreibung

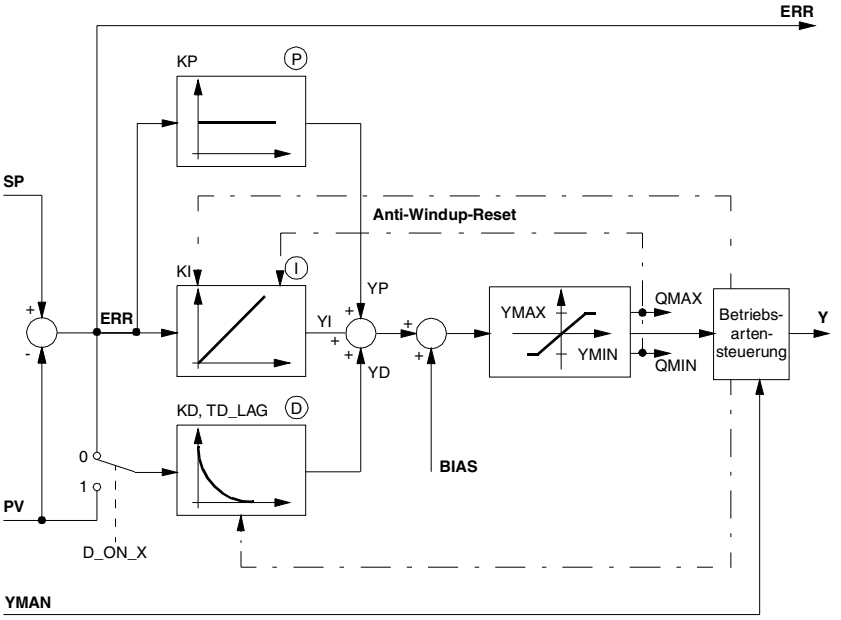
Beschreibung der Bausteinparameter

Parameter	Datentyp	Bedeutung
MAN	BOOL	"1": Betriebsart Hand
HALT	BOOL	"1": Betriebsart Halt
SP	REAL	Sollwerteingang
PV	REAL	Eingangsgröße
BIAS	REAL	Störgrößeneingang
D_ON_X	BOOL	"1": D-Anteil bezogen auf Regelgröße; "0": D-Anteil bezogen auf Regeldifferenz
REVERS	BOOL	"1": Ausgang umgekehrt
KP	REAL	Proportionalbeiwert (Verstärkungsfaktor)
KI	REAL	Integrierbeiwert
KD	REAL	Differenzierbeiwert
TD_LAG	TIME	Verzögerungszeit, D-Anteil
YMAX	REAL	Obere Stellgrenze
YMIN	REAL	Untere Stellgrenze
YMAN	REAL	Hand-Stellwert
Y	REAL	Stellgröße
ERR	REAL	Regeldifferenz
QMAX	BOOL	"1" = Y hat obere Stellgrenze erreicht
QMIN	BOOL	"1" = Y hat untere Stellgrenze erreicht

Parametrierung des PIDP1-Reglers

Strukturbild

Nachfolgend finden Sie das Strukturbild des Bausteins PIDP1:



Parametrierung

Die Struktur des PIDP1-Reglers ist im *Strukturbild, S. 383* dargestellt.

Die Parametrierung des PIDP1-Reglers erfolgt zunächst durch die reinen PID-Parameter, nämlich den Proportionalbeiwert KP, den Integrierbeiwert KI und den Differenzierbeiwert KD.

Der P-, I- und D-Anteil können einzeln abgeschaltet werden, indem der entsprechende Eingang (KP, KI oder KD) auf 0 gesetzt wird.

Der D-Anteil wird mit der Zeitkonstante TD_LAG verzögert. Der D-Anteil kann entweder durch die Regeldifferenz ERR (D_ON_X =0) oder durch die Regelgröße PV (D_ON_X = 1) gebildet werden. Wird der D-Anteil durch die Regelgröße PV bestimmt, so wird bei Führungsgrößenänderungen (Änderungen im Eingang SP) kein Sprung bedingt durch den D-Anteil entstehen. Der D-Anteil wirkt sich im Prinzip nur auf Störungen bzw. Prozeßänderungen aus.

Umkehr des Regelsinns

Ein entgegengesetztes Verhalten des Reglers kann durch Setzen des Eingangs REVERS auf 1 erreicht werden. REVERS = 0 bewirkt, daß bei einer positiven Störgröße der Ausgangswert steigt. REVERS = 1 bewirkt, daß bei einer positiven Störgröße der Ausgangswert fällt.

Stellgrößen-Begrenzung

Die Grenzen YMAX und YMIN begrenzen den Ausgang nach oben sowie nach unten. Damit gilt YMIN ≤ Y ≤ YMAX.

Das Erreichen der Grenzwerte, bzw. eine Begrenzung des Ausgangssignals wird durch die beiden Merker QMAX und QMIN angezeigt:

- QMAX = 1 wenn Y ≥ YMAX
- QMIN = 1 wenn Y ≤ YMIN

Die Obergrenze YMAX für die Begrenzung der Stellgröße ist größer als die Untergrenze YMIN zu wählen.

Anti-Windup-Reset

Findet eine Stellgrößen-Begrenzung statt, soll durch Anti-Windup-Reset dafür gesorgt werden, daß der Integrale Anteil "nicht über alle Grenzen wachsen kann". Die Anti-Windup-Maßnahme wird nur dann durchgeführt, wenn der I-Anteil des Reglers nicht abgeschaltet ist. Die Grenzen für Anti-Windup sind hier dieselben, wie für die Stellgrößenbegrenzung. Der D-Anteil wird für die Anti-Windup-Maßnahme nicht mit berücksichtigt, so daß Spitzen, hervorgerufen durch den D-Anteil, nicht durch die Anti-Windup-Maßnahme gekappt werden.

Die Anti-Windup-Maßnahme korrigiert den I-Anteil in der Form, daß gilt:

$$YMIN - YP - BIAS \leq YI \leq YMAX - YP - BIAS$$

Auswahl der Reglertypen

Es gibt verschiedene Reglertypen, die über die Parameter KP, KI und KD ausgewählt werden:

Reglertyp	KP	KI	KD
P-Regler	> 0	= 0	= 0
PI-Regler	> 0	> 0	= 0
PD-Regler	> 0	= 0	> 0
PID-Regler	> 0	> 0	> 0
I-Regler	= 0	> 0	= 0

Betriebsarten

Auswahl der Betriebsarten

Es gibt drei Betriebsarten, die über die Parameter MAN und HALT ausgewählt werden:

Betriebsart	MAN	HALT
Automatik	0	0
Hand	1	0 oder 1
Halt	0	1

Betriebsart Automatik

n der Betriebsart Automatik wird der Stellausgang Y durch den diskretisierten PID-Algorithmus, in Abhängigkeit von Regelgröße PV und Führungsgröße SP bestimmt. Der Stellausgang wird durch YMAX und YMIN begrenzt. Die Stellgrenzen sind gleichzeitig auch Grenzen für den Antiwindup-Reset.

Die Umschaltung von Automatik nach Hand ist normalerweise nicht stoßfrei, da der Ausgang Y jeden Wert zwischen YMAX und YMIN haben kann und Y bei der Umschaltung direkt den Wert YMAN annimmt.

Soll die Umschaltung von Automatik nach Hand trotzdem stoßfrei erfolgen, gibt es zwei Möglichkeiten, die für einen PID1-Regler (siehe *Umschaltung von Automatik nach Hand, S. 326*) beispielhaft erklärt sind.

Betriebsart Hand

In der Betriebsart Hand wird der Handstellwert YMAN fest auf den Stellausgang Y übertragen. Der Stellausgang wird jedoch durch YMAX und YMIN begrenzt. Die internen Größen werden so nachgeführt, daß der Regler (bei zugeschaltetem I-Anteil) stoßfrei von Hand auf Automatik umgeschaltet werden kann. Die Stellgrenzen sind gleichzeitig auch Grenzen für den Antiwindup-Reset.

Der D-Anteil wird in dieser Betriebsart automatisch auf 0 gesetzt.

Betriebsart Halt

In der Betriebsart Halt bleibt der Reglerausgang, wie er vorgefunden wird, der Funktionsbaustein verändert den Reglerausgang Y nicht (Regler bleibt stehen), d. h. Y = Y(old). Die internen Größen werden so nachgeführt, daß die Summe ihrer Anteile dem Stellausgang entsprechen, so daß der Regler (bei zugeschaltetem I-Anteil) ohne Stoß von seiner aktuellen Lage aus weiterfährt. Die Stellgrenzen sind gleichzeitig auch Grenzen für den Antiwindup-Reset.

Der D-Anteil wird in dieser Betriebsart automatisch auf 0 gesetzt.

Detaillierte Formeln

Erläuterung der Formelgrößen

Bedeutung der Größen in den Formeln:

Größe	Bedeutung
dt	Zeitdifferenz zwischen aktuellem Zyklus und vorherigem Zyklus
ERR	Regeldifferenz (SP - PV)
$ERR_{(new)}$	Wert der Regeldifferenz aus dem aktuellen Abtastschritt
$ERR_{(old)}$	Wert der Regeldifferenz aus dem vorherigen Abtastschritt
BIAS	Störgröße
$PV_{(new)}$	Wert der Regelgröße aus dem aktuellen Abtastschritt
$PV_{(old)}$	Wert der Regelgröße aus dem vorherigen Abtastschritt.
Y	aktueller Ausgang (Betriebsart Halt) oder YMAN (Betriebsart Hand)
YD	D-Anteil
YI	I-Anteil
YP	P-Anteil

Stellgröße

Die Stellgröße setzt sich aus verschiedenen Teilgrößen zusammen:

$$Y = YP + YI + YD + BIAS$$

Nach der Summation der Anteile findet eine Stellgrößen-Begrenzung statt, so daß gilt:

$$YMIN \leq Y \leq YMAX$$

Regeldifferenz

Die Regeldifferenz wird wie folgt ermittelt:

Wenn...	Dann ...
REVERS = 0	ERR = SP - PV
REVERS = 1	ERR = PV - SP

Übersicht zur Berechnung der Regler-Anteile

Nachfolgend finden Sie eine Übersicht über die unterschiedliche Berechnung der Regler-Anteile in Abhängigkeit von den Verstärkungsfaktoren KP, KI und KD:

- P-Anteil YP für Betriebsart Hand, Halt und Automatik
- I-Anteil YI für Betriebsart Automatik
- I-Anteil YI für Betriebsart Hand und Halt
- D-Anteil YD für Betriebsart Automatik
- D-Anteil YD für Betriebsart Hand und Halt

P-Anteil YP für alle Betriebsarten

YP für Betriebsart Hand, Halt und Automatik wird wie folgt ermittelt:

$$YP = KP \times ERR$$

I-Anteil YI für Betriebsart Automatik

YI für Betriebsart Automatik wird wie folgt ermittelt:

Für KI > 0 gilt:

$$YI_{(new)} = YI_{(old)} + KI \times dt \times \frac{ERR_{(new)} + ERR_{(old)}}{2}$$

Für KI = 0 gilt:

$$YI = 0$$

Der I-Anteil wird nach der Trapezregel gebildet.

I-Anteil YI für Betriebsart Hand und Halt

YI für Betriebsart Hand und Halt wird wie folgt ermittelt:

Für KI > 0 gilt:

$$YI = Y - (YP - BIAS)$$

Für KI = 0 gilt:

$$YI = 0$$

D-Anteil YD für Betriebsart Automatik

YD für Betriebsart Automatik und Kaskade wird wie folgt ermittelt:

Für KD > 0 und D_ON_X = 0 gilt:

$$YD_{(new)} = \frac{TD_LAG}{dt + TD_LAG} \times (YD_{(old)} + KD \times (ERR_{(new)} - ERR_{(old)}))$$

Für KD > 0 und D_ON_X = 1 gilt:

$$YD_{(new)} = \frac{TD_LAG}{dt + TD_LAG} \times (YD_{(old)} + KD \times (PV_{(old)} - PV_{(new)}))$$

Für KD = 0 gilt:

$$YD = 0$$

D-Anteil YD für Betriebsart Hand und Halt

YD für Betriebsart Hand und Halt wird wie folgt ermittelt:

$$YD = 0$$

Laufzeitfehler

Fehlermeldung Bei YMAX < YMIN erfolgt eine Fehlermeldung.

PIP: PIP-Kaskaden-Regler

41

Übersicht

Einleitung

Diese Kapitel beschreibt den Baustein PIP.

Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Kurzbeschreibung	390
Darstellung	391
Strukturbild des Funktionsbausteins PIP	393
Parametrierung des PIP-Kaskaden-Reglers	394
Betriebsarten	396
Detaillierte Formeln	398
Laufzeitfehler	399

Kurzbeschreibung

Funktionsbeschreibung

Der Funktionsbaustein stellt einen Kaskaden-Regler, bestehend aus einem PI-Führungsregler und einem P-Folgeregler, dar.

Zwischen Führungsgröße SP und der Regelgröße PV wird die Regeldifferenz ERR gebildet.

Der Führungsregler erzeugt durch diese Regeldifferenz einen Sollwert SP2 für den Folgeregler. Aufgrund der Differenz von SP2 und PV2 erzeugt der Folgeregler die Stellgröße Y.

Als zusätzliche Parameter können EN und ENO projektiert werden.

Eigenschaften

Der Funktionsbaustein besitzt folgende Eigenschaften:

- PI als Führungsregler, P als Folgeregler
- Stellgrößen-Begrenzung
- Anti-Windup-Reset für den PI-Regler
- Betriebsarten Hand, Festwert-Regelung, Halt, Automatik

Übertragungsfunktion

Die Übertragungsfunktionen für die Regler lauten:

Regler	Übertragungsfunktion
Führungsregler (PI-Regler)	$G(s) = gain1 \times \left(1 + \frac{1}{ti \times s}\right)$
Folgeregler (P-Regler)	$G(s) = gain2$

Proportionalbeiwert

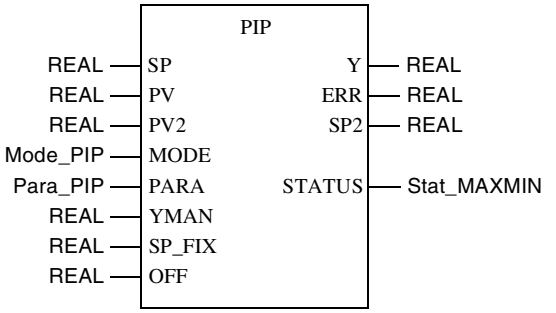
Der Proportionalbeiwert des Führungsreglers wird wie folgt ermittelt:

$YP = gain1 \times ERR$

Darstellung

Symbol

Darstellung des Bausteins



Parameterbeschreibung PIP

Beschreibung der Bausteinparameter

Parameter	Datentyp	Bedeutung
SP	REAL	Führungsgröße
PV	REAL	Regelgröße für den Führungsregler
PV2	REAL	Regelgröße für den Folgeregler (Hilfsregelgröße)
MODE	Mode_PIP	Betriebsarten
PARA	Para_PIP	Parameter
YMAN	REAL	Handwert (vom Ausgang Y)
SP_FIX	REAL	Festwert (= Handwert der Führungsgröße vom Folgeregler)
OFF	REAL	Offset am Ausgang des P-Reglers
Y	REAL	Stellgröße
ERR	REAL	Regeldifferenz
SP2	REAL	Sollwert für Folgeregler
STATUS	Stat_MAXMIN	Status des Ausgangs Y

Parameterbeschreibung Mode_PIP

Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
man	BOOL	"1": Betriebsart Hand
halt	BOOL	"1": Betriebsart Halt
fix	BOOL	"1": Betriebsart Festwert-Regelung

Parameterbeschreibung Para_PIP

Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
gain1	REAL	Proportionalbeiwert (Verstärkungsfaktor) für PI-Regler
ti	TIME	Nachstellzeit für PI-Regler
gain2	REAL	Proportionalbeiwert (Verstärkungsfaktor) für P-Regler
ymax	REAL	Obere Stellgrenze
ymin	REAL	Untere Stellgrenze

Parameterbeschreibung Stat_MAXMIN

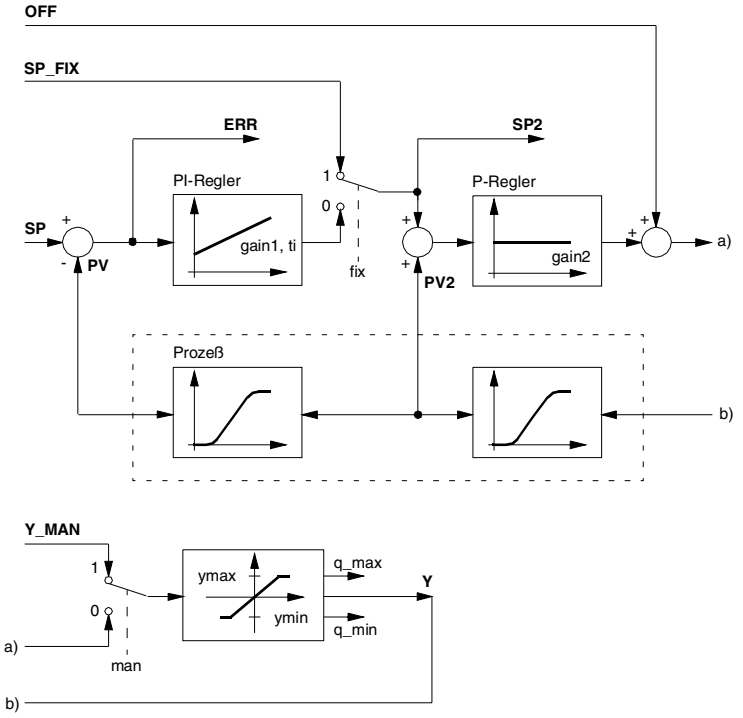
Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
qmax	BOOL	"1" = Y hat obere Stellgrenze erreicht
qmin	BOOL	"1" = Y hat untere Stellgrenze erreicht

Strukturbild des Funktionsbausteins PIP

Strukturbild

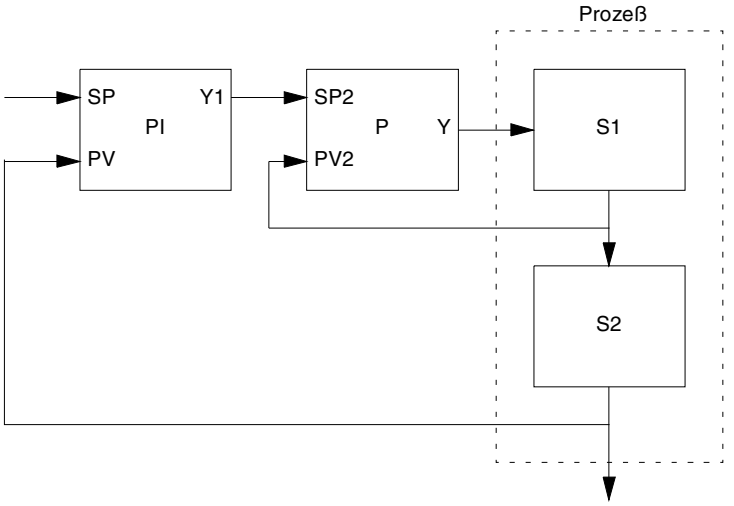
Nachfolgend finden Sie das Strukturbild des Bausteins PIP:



Parametrierung des PIP-Kaskaden-Reglers

Blockschaltbild

Blockschaltbild des PIP-Kaskaden-Regler



Parametrierung

Die Struktur des PIP-Reglers ist im *Blockschaltbild, S. 394* dargestellt.

Die Parametrierung des Funktionsbausteins erfolgt zunächst durch die reinen PI-Parameter, nämlich den Proportionalbeiwert gain1 und der Nachstellzeit ti.

Der I-Anteil kann durch Nullsetzen von ti abgeschaltet werden.

Anschließend folgt die Parametrierung des P-Reglers durch den Proportionalbeiwert gain2.

Stellgrößen-Begrenzung

Am Ausgang des Folgereglers findet eine Stellgrößen-Begrenzung statt, so daß gilt:

$ymin \leq Y \leq ymax$

Anti-Windup-Reset (PI-Regler)

Findet eine Stellgrößen-Begrenzung statt, soll durch Antiwindup-Reset dafür gesorgt werden, daß der integrale Anteil des Führungsreglers "nicht über alle Grenzen wachsen kann". Die Antiwindup-Maßnahme wird nur dann durchgeführt, wenn der I-Anteil des Reglers nicht abgeschaltet ist (ti = 0).

Die Antiwindup-Grenzen für den PI-Führungsregler werden dynamisch an die aktuelle Regeldifferenz des Folgereglers und die Grenzen ymax und ymin angepaßt.

Findet eine Stellgrößen-Begrenzung statt, wird der Integralanteil wie folgt begrenzt:

- bei Erreichen der oberen Grenze:

$$YI = \left(\frac{ymax - OFF}{gain2} + PV\right) - YP$$

- bei Erreichen der unteren Grenze:

$$YI = \left(\frac{ymin - OFF}{gain2} + PV\right) - YP$$

Betriebsarten

Auswahl der Betriebsarten

Es gibt vier Betriebsarten, die über die Elemente man, halt und fix ausgewählt werden:

Betriebsart	man	halt	fix
Automatik	0	0	0
Hand	1	0 oder 1	0
Halt	0	1	0
Festwert-Regelung	0	0	1

Betriebsart Automatik

In der Betriebsart Automatik wird der Stellausgang Y durch die Regelung bestimmt, in Abhängigkeit von den Regelgrößen PV, PV2 und den Führungsgrößen SP, SP2. Der Stellausgang wird durch ymax und ymin begrenzt.

Die Umschaltung von Automatik nach Hand ist normalerweise nicht stoßfrei, da der Ausgang Y jeden Wert zwischen ymax und ymin haben kann und Y bei der Umschaltung direkt den Wert YMAN annimmt.

Soll die Umschaltung von Automatik nach Hand trotzdem stoßfrei erfolgen, gibt es zwei Möglichkeiten, die für einen PID-Regler (siehe *Umschaltung von Automatik nach Hand, S. 312*) beispielhaft erklärt sind.

Betriebsart Hand

In der Betriebsart Hand arbeitet der P-Regler in der Betriebsart Hand. Der I-Anteil des PI-Reglers wird für stoßfreies Umschalten nachgeführt.

Es wird der Handstellwert YMAN fest auf den Stellausgang Y übertragen. Der Stellausgang wird jedoch durch ymax und ymin begrenzt. Der Integralanteil des Führungsreglers wird so nachgeführt, daß der Regler (bei zugeschaltetem I-Anteil) stoßfrei von Hand auf Automatik umgeschaltet werden kann.

Betriebsart Halt

In der Betriebsart Halt bleibt der Reglerausgang, wie er vorgefunden wird, der Funktionsbaustein verändert den Reglerausgang Y nicht (Regler bleibt stehen), d.h. Y = Y(old). Die Betriebsart Halt eignet sich unter anderem auch dafür, den Reglerausgang Y über ein externes Bediengerät zu stellen. Dabei werden die internen Anteile im Regler richtig nachgeführt, sodaß der Regler ohne Stoß von seiner aktuellen Lage aus weiterfährt. Der Stellausgang wird jedoch durch ymax und ymin begrenzt.

Betriebsart Festwert-Regelung

In der Betriebsart Festwert-Regelung arbeitet der P-Regler in der Betriebsart Automatik und der PI-Regler in der Betriebsart Halt.

Es wird der Festwert SP_FIX fest auf den Stellausgang des PI-Reglers Y1 (= SP2) übertragen. Der Stellausgang des PIP-Reglers Y wird durch ymax und ymin begrenzt. Der Intergralanteil des Führungsreglers wird so nachgeführt, daß der Regler (bei zugeschaltetem I-Anteil) stoßfrei von Festwertregelung auf Automatik umgeschaltet werden kann.

Detaillierte Formeln

Erläuterung der Formelgrößen

Bedeutung der Größen in den nachfolgenden Formeln:

Größe	Bedeutung
dt	Zeitdifferenz zwischen aktuellem Zyklus und vorherigem Zyklus
ERR	Regeldifferenz (SP - PV)
$ERR_{(new)}$	Wert der Regeldifferenz aus dem aktuellen Abtastschritt
$ERR_{(old)}$	Wert der Regeldifferenz aus dem vorherigen Abtastschritt
OFF	Offset am Ausgang des P-Reglers
Y	Stellgröße
Y1	Y des Hauptreglers
YI	I-Anteil
YP	P-Anteil

Übersicht zur Berechnung der Regler-Anteile

Nachfolgend finden Sie eine Übersicht über die unterschiedliche Berechnung der Regler-Anteile und Ausgänge für die verschiedenen Betriebsarten:

- YI, Y, SP2 in der Betriebsart Automatik
- YI, Y und SP2 in der Betriebsart Hand
- YI, Y und SP2 in der Betriebsart Halt
- YI, Y und SP2 in der Betriebsart Festwert-Regelung

Betriebsart Automatik

Das Ausgangssignal Y des Kaskadenreglers ergibt sich zu:

$$Y = (SP2 - PV2) \times gain2 + OFF$$

Das Eingangssignal SP2 des Folgereglers ergibt sich zu:

$$SP2 = YP + YI$$

Der Integralanteil YI des Führungsreglers für die Betriebsart Automatik wird wie folgt ermittelt:

$$YI_{(new)} = YI_{(old)} + gain1 \times \frac{dt}{ti} \times \frac{ERR_{(new)} + ERR_{(old)}}{2}$$

Der I-Anteil wird nach der Trapezregel gebildet.

Betriebsart Hand

Das Ausgangssignal Y des Kaskadenreglers ergibt sich zu:

$$Y = YMAN$$

Das Eingangssignal SP2 des Folgereglers ergibt sich zu:

$$SP2 = \frac{Y - OFF}{gain2} + PV2$$

Der Integralanteil YI des Führungsreglers für die Betriebsart Hand wird wie folgt ermittelt:

$$YI = SP2 - (SP - PV) \times gain1$$

Betriebsart Halt

Das Ausgangssignal Y des Kaskadenreglers ergibt sich zu:

$$Y = Y_{(old)}$$

Das Eingangssignal SP2 des Folgereglers ergibt sich zu:

$$SP2 = \frac{Y - OFF}{gain2} + PV2$$

Der Integralanteil YI des Führungsreglers für die Betriebsart Halt wird wie folgt ermittelt:

$$YI = SP2 - (SP - PV) \times gain1$$

Betriebsart Festwert-Regelung

Das Ausgangssignal Y des Kaskadenreglers ergibt sich zu:

$$Y = (SP2 - PV2) \times gain2 + OFF$$

Das Eingangssignal SP2 des Folgereglers ergibt sich zu:

$$SP2 = SP_FIX$$

Der Integralanteil YI des Führungsreglers für die Betriebsart Festwert-Regelung wird wie folgt ermittelt:

$$YI = SP2 - (SP - PV) \times gain1$$

Laufzeitfehler

Fehlermeldung

Es erfolgt eine Fehlermeldung, wenn

- eine unzulässige Gleitkommazahl am Eingang PV, PV2, YMAN oder SP_FIX anliegt
- ymax < ymin ist.

PPI: PPI-Kaskaden-Regler

42

Übersicht

Einleitung

Diese Kapitel beschreibt den Baustein PPI.

Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Kurzbeschreibung	402
Darstellung	403
Strukturbild des Funktionsbausteins PPI	405
Parametrierung des PPI-Kaskaden-Reglers	406
Betriebsarten	408
Detaillierte Formeln	409
Laufzeitfehler	410

Kurzbeschreibung

Funktionsbeschreibung

Der Funktionsbaustein stellt einen Kaskaden-Regler, bestehend aus einem P-Führungsregler und einem PI-Folgeregler, dar.

Zwischen Führungsgröße SP und der Regelgröße PV wird die Regeldifferenz ERR gebildet.

Der Führungsregler erzeugt durch diese Regeldifferenz einen Sollwert SP2 für den Folgeregler. Aufgrund der Differenz von SP2 und PV2 erzeugt der Folgeregler die Stellgröße Y.

Als zusätzliche Parameter können EN und ENO projektiert werden.

Eigenschaften

Der Funktionsbaustein besitzt folgende Eigenschaften:

- P als Führungsregler, PI als Folgeregler
- Stellgrößen-Begrenzung
- Anti-Windup-Reset für den PI-Regler
- Betriebsarten Hand, Festwert-Regelung, Halt, Automatik

Übertragungsfunktion

Die Übertragungsfunktionen für die Regler lauten:

Regler	Übertragungsfunktion
Führungsregler (P-Regler)	$G(s) = gain1$
Folgeregler (PI-Regler)	$G(s) = gain2 \times \left(1 + \frac{1}{ti \times s}\right)$

Proportionalbeiwert

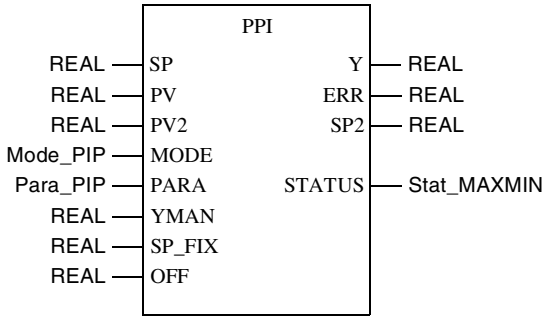
Der Proportionalbeiwert des Folgereglers wird wie folgt ermittelt:

$$YP = gain2 \times (SP2 - PV2)$$

Darstellung

Symbol

Darstellung des Bausteins



Parameterbeschreibung PPI

Beschreibung der Bausteinparameter

Parameter	Datentyp	Bedeutung
SP	REAL	Führungsgröße für den Führungsregler
PV	REAL	Regelgröße für den Führungsregler
PV2	REAL	Regelgröße für den Folgeregler (Hilfsregelgröße)
MODE	Mode_PPI	Betriebsarten
PARA	Para_PPI	Parameter
YMAN	REAL	Handwert (vom Ausgang Y)
SP_FIX	REAL	Festwert (= Handwert der Führungsgröße vom Folgeregler)
OFF	REAL	Offset am Ausgang des P-Reglers
Y	REAL	Stellgröße
ERR	REAL	Regeldifferenz
SP2	REAL	Sollwert für Folgeregler
STATUS	Stat_MAXMIN	Status des Ausgangs Y

Parameterbeschreibung Mode_PPI

Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
man	BOOL	"1": Betriebsart Hand
halt	BOOL	"1": Betriebsart Halt
fix	BOOL	"1": Betriebsart Festwert-Regelung

Parameterbeschreibung Para_PPI

Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
gain1	REAL	Proportionalbeiwert (Verstärkungsfaktor) für P-Regler
ti	TIME	Nachstellzeit für PI-Regler
gain2	REAL	Proportionalbeiwert (Verstärkungsfaktor) für PI-Regler
ymax	REAL	Obere Stellgrenze
ymin	REAL	Untere Stellgrenze

Parameterbeschreibung Stat_MAXMIN

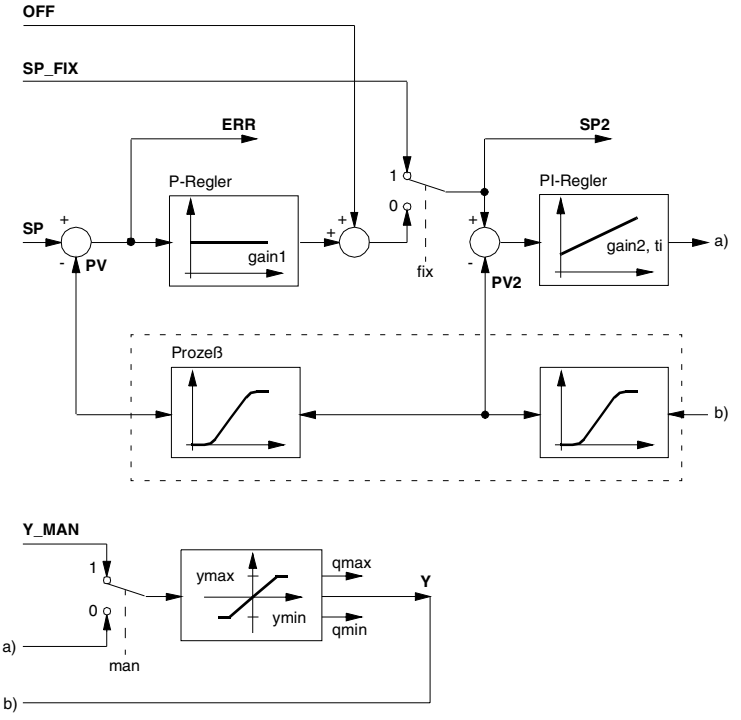
Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
qmax	BOOL	"1" = Y hat obere Stellgrenze erreicht
qmin	BOOL	"1" = Y hat untere Stellgrenze erreicht

Strukturbild des Funktionsbausteins PPI

Strukturbild

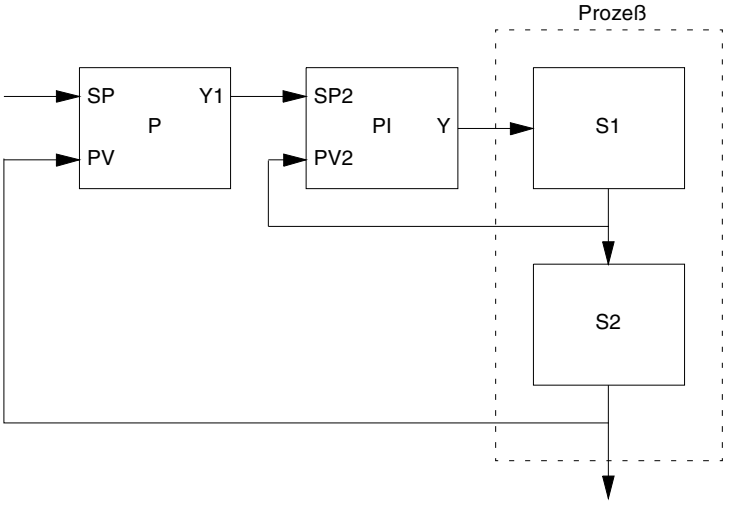
Nachfolgend finden Sie das Strukturbild des Bausteins PPI:



Parametrierung des PPI-Kaskaden-Reglers

Blockschaltbild

Blockschaltbild des PPI-Kaskaden-Regler



Parametrierung

Die Struktur des PPI-Reglers ist im *Blockschaltbild, S. 406* dargestellt.

Die Parametrierung des Funktionsbausteins erfolgt zunächst durch den Proportionalbeiwert gain1 und den Offset für den Ausgang des P-Reglers (OFF).

Anschließend folgt die Parametrierung des PI-Reglers durch den Proportionalbeiwert gain2 und die Nachstellzeit ti.

Der I-Anteil kann durch Nullsetzen von ti abgeschaltet werden.

Die Grenzen ymax und ymin begrenzen den Ausgang nach oben sowie nach unten.

Das Erreichen der Grenzwerte, bzw. eine Begrenzung des Ausgangssignals wird durch die beiden Merker qmax und qmin angezeigt:

- qmax = 1, wenn $Y \geq ymax$
- qmin = 1, wenn $Y \leq ymin$

Stellgrößen-Begrenzung

Nach der Summation der Anteile findet eine Stellgrößen-Begrenzung statt, so daß gilt: $ymin \leq Y \leq ymax$

Anti-Windup-Reset (PI-Regler)

Findet eine Stellgrößen-Begrenzung statt, soll durch Antiwindup-Reset dafür gesorgt werden, daß der integrale Anteil des Führungsreglers "nicht über alle Grenzen wachsen kann". Die Antiwindup-Maßnahme wird nur dann durchgeführt, wenn der I-Anteil des Folgereglers nicht abgeschaltet ist.

Der Antiwindup-Reset findet statt, wenn:

$Y \geq ymax$ oder $Y \leq ymin$

In diesem Fall gilt:

$YI = Y - YP$

Betriebsarten

Auswahl der Betriebsarten

Es gibt vier Betriebsarten, die über die Elemente man, halt und fix ausgewählt werden:

Betriebsart	man	halt	fix
Automatik	0	0	0
Hand	1	0 oder 1	0
Halt	0	1	0
Festwert-Regelung	0	0	1

Betriebsart Automatik

In der Betriebsart Automatik wird der Stellausgang Y durch die Regelung bestimmt, in Abhängigkeit von den Regelgrößen PV, PV2 und den Führungsgrößen SP, SP2. Der Stellausgang wird durch ymax und ymin begrenzt.

Die Umschaltung von Automatik nach Hand ist normalerweise nicht stoßfrei, da der Ausgang Y jeden Wert zwischen ymax und ymin haben kann und Y bei der Umschaltung direkt den Wert YMAN annimmt.

Soll die Umschaltung von Automatik nach Hand trotzdem stoßfrei erfolgen, gibt es zwei Möglichkeiten, die für einen PID-Regler (siehe *Umschaltung von Automatik nach Hand, S. 312*) beispielhaft erklärt sind.

Betriebsart Hand

Es wird der Handstellwert YMAN fest auf den Stellausgang Y übertragen. Der Stellausgang wird jedoch durch ymax und ymin begrenzt. Die internen Größen werden so nachgeführt, daß der Regler (bei zugeschaltetem I-Anteil) stoßfrei von Hand auf Automatik umgeschaltet werden kann.

Betriebsart Halt

In der Betriebsart Halt bleibt der Reglerausgang, wie er vorgefunden wird, der Funktionsbaustein verändert den Reglerausgang Y nicht (Regler bleibt stehen), d.h. Y = Y(old). Die Betriebsart Halt eignet sich unter anderem auch dafür, den Reglerausgang Y über ein externes Bediengerät zu stellen. Dabei werden die internen Anteile im Regler richtig nachgeführt, sodaß der Regler ohne Stoß von seiner aktuellen Lage aus weiterfährt. Der Stellausgang wird jedoch durch ymax und ymin begrenzt.

Betriebsart Festwert-Regelung

In dieser Betriebsart wird der Festwert SP_FIX auf den Sollwerteingang des PI-Reglers (SP2) übertragen. Der PI-Regler arbeitet in der Betriebsart Automatik.

Detaillierte Formeln

Erläuterung der Formelgrößen

Bedeutung der Größen in den nachfolgenden Formeln:

Größe	Bedeutung
dt	Aktuelle Abtastzeit
ERR	Regeldifferenz (SP - PV)
$err2_{(new)}$	Regeldifferenz (SP2-PV2)
$err2_{(old)}$	Wert der Regeldifferenz aus dem vorherigen Abtastschritt
OFF	Offset am Ausgang des P-Reglers
Y	Stellgröße
YI	I-Anteil
YP	P-Anteil

Ausgang des Führungsreglers

Der Ausgang des Führungsreglers wird wie folgt ermittelt:

$$Y1 = SP2 = gain1 \times ERR + OFF$$

Übersicht zur Berechnung der Regler-Anteile

Nachfolgend finden Sie eine Übersicht der unterschiedlichen Berechnungen der Regler-Anteile und Ausgänge in Abhängigkeit von den verschiedenen Betriebsarten:

- YI und Y in der Betriebsart Automatik
- YI, Y und SP2 in der Betriebsart Hand
- YI, Y und SP2 in der Betriebsart Halt
- YI, YP, Y und SP2 in der Betriebsart Festwert-Regelung

Betriebsart Automatik

Das Ausgangssignal Y des Kaskadenreglers ergibt sich zu:

$$Y = YP + YI$$

Der Integralanteil YI des Folgereglers für die Betriebsart Automatik wird wie folgt ermittelt:

$$YI_{(new)} = YI_{(old)} + gain2 \times \frac{dt}{ti} \times \frac{err2_{(new)} + err2_{(old)}}{2}$$

Der I-Anteil wird nach der Trapezregel gebildet.

Betriebsart Hand

Das Ausgangssignal Y des Kaskadenreglers ergibt sich zu:

$$Y = YMAN$$

Das Eingangssignal SP2 des Folgereglers ergibt sich zu:

$$SP2 = gain1 \times (SP - PV) + OFF$$

Der Integralanteil YI des Folgereglers für die Betriebsart Hand wird wie folgt ermittelt:

$$YI = Y - (SP2 - PV2) \times gain2$$

Betriebsart Halt

Das Ausgangssignal Y des Kaskadenreglers ergibt sich zu:

$$Y = Y_{(old)}$$

Das Eingangssignal SP2 des Folgereglers ergibt sich zu:

$$SP2 = gain1 \times (SP - PV) + OFF$$

Der Integralanteil YI des Folgereglers für die Betriebsart Halt wird wie folgt ermittelt:

$$YI = Y - (SP2 - PV2) \times gain2$$

Betriebsart Festwert-Regelung

Das Ausgangssignal Y des Kaskadenreglers ergibt sich zu:

$$Y = YP + YI$$

Das Eingangssignal SP2 des Folgereglers ergibt sich zu:

$$SP2 = SP_FIX$$

Der Integralanteil YI des Folgereglers für die Betriebsart Festwert-Regelung wird wie folgt ermittelt:

$$YI_{(new)} = YI_{(old)} + gain2 \times \frac{dt}{ti} \times \frac{err2_{(new)} + err2_{(old)}}{2}$$

Der Proportionalanteil YP wird wie folgt ermittelt:

$$YP = gain2 \times (SP2 - PV2)$$

Laufzeitfehler

Fehlermeldung

Es erfolgt eine Fehlermeldung, wenn

- eine unzulässige Gleitkommazahl am Eingang PV, PV2, YMAN oder SP_FIX anliegt
- ymax < ymin ist.

PWM: Pulsbreitenmodulation

43

Übersicht

Einleitung

Dieses Kapitel beschreibt den Baustein PWM.

Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Kurzbeschreibung	412
Darstellung	413
Formeln	414
Detailbeschreibung	415
Beispiele zum Baustein PWM	418

Kurzbeschreibung

Verwendung des Bausteins

Die Ansteuerung von Stellgliedern erfolgt nicht nur über analoge Größen sondern auch mittels binärer Stellsignale. Die Wandlung von analogen Werten in binäre Ausgangssignale erfolgt beispielsweise über Pulsbreitenmodulation (PWM) oder Pulsdauermodulation (PDM).

Dabei soll die eingestellte mittlere Energie des Stellgliedes (Stellenergie) dem analogen Eingangswert (X) des Modulationsbausteins entsprechen.

Funktionsbeschreibung

Der Funktionsbaustein PWM dient zur Umsetzung von analogen Werten in digitale Ausgangssignale für Concept.

Bei der Pulsbreitenmodulation (PWM) wird bei konstanter Taktperiode ein in seiner Dauer vom Analogwert abhängiges 1-Signal ausgegeben. Die mittlere eingestellte Energie entspricht dem Quotienten aus der Einschaltdauer T_on und der Taktzeit t_period.

Damit die mittlere eingestellte Energie auch der analogen Eingangsgröße X entspricht muß folgendes gelten:

$$T_on \sim X$$

Als zusätzliche Parameter können EN und ENO projektiert werden.

Allgemeines zur Stellgliedansteuerung

Im allgemeinen wird die binäre Stellgliedansteuerung durch zwei binäre Signale Y_POS und Y_NEG vorgenommen.

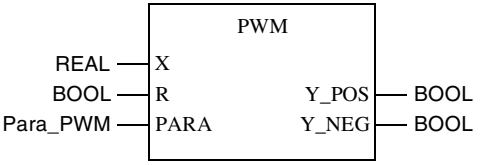
Bei einem Motor entspricht der Ausgang Y_POS beispielsweise dem Signal "Rechtslauf" und der Ausgang Y_NEG dem Signal "Linkslauf". Bei einem Ofen können die Ausgänge Y_POS und Y_NEG entsprechend als "Heizen" und "Kühlen" interpretiert werden.

Handelt es sich beim Stellantrieb um einen Motor, so muß unter Umständen bei nicht selbsthemmenden Getrieben, zur Vermeidung des Nachlaufens, nach dem Einschaltsignal ein Bremsimpuls ausgegeben werden. Zum Schutz der Leistungselektronik muß nach dem Einschaltimpuls T_on vor dem Bremsimpuls t_brake eine Pausenzeit t_pause eingelegt werden, um Kurzschlüsse zu verhindern.

Darstellung

Symbol

Darstellung des Bausteins



Parameterbeschreibung PWM

Beschreibung der Bausteinparameter

Parameter	Datentyp	Bedeutung
X	REAL	Eingangsgröße
R	BOOL	Betriebsart Reset ("1" = Reset)
PARA	Para_PWM	Parameter
Y_POS	BOOL	Ausgang für positive X-Werte
Y_NEG	BOOL	Ausgang für negative X-Werte

Parameterbeschreibung Para_PWM

Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
t_period	TIME	Periodendauer
t_pause	TIME	Pausenzeit
t_brake	TIME	Bremszeit
t_min	TIME	Minimalzeit für Stellimpuls
t_max	TIME	Maximalzeit für Stellimpuls
up_pos	REAL	Oberer Grenzwert für positive X-Werte
up_neg	REAL	Oberer Grenzwert für negative X-Werte

Formeln

Impulslänge für Y_POS und Y_NEG

Die Impulslänge T_on für die Ausgänge Y_POS und Y_NEG wird nach folgenden Gleichungen bestimmt:

Ausgang	Formel	Bedingung
Y_POS	$T_on = t_period \times \frac{X}{up_pos}$	$0 \le X \le up_pos$
Y_NEG	$T_on = t_period \times \frac{\lvert X \rvert}{up_neg}$	$up_neg \le -X \le 0$

Regeln zur Parametrierung

Für einen ordnungsgemäßen Betrieb sind bei der Parametrierung folgende Regeln zu beachten:

- $(2 \times t_pause + t_brake + t_max) \le t_period$
- Von den Parametern up_pos und up_neg wird jeweils der Betrag ausgewertet.

Detailbeschreibung

Arbeitsweise des Bausteins

Die Periodendauer legt die Zeit fest, in welcher die Stellimpulse (1-Signal am Ausgang Y_POS bzw. Y_NEG) regelmäßig d.h. im konstanten Zeitraster ausgegeben werden.

Der Parameter t_min gibt die minimale Impulslänge an, d.h. die Zeitdauer, die der Ausgang Y_POS bzw. der Ausgang Y_NEG minimal 1-Signal besitzen soll. Ist die nach der Gleichung im Abschnitt "*Formeln, S. 414*" berechnete Impulsdauer kleiner als t_min, so wird für die gesamte Periode kein Impuls ausgegeben.

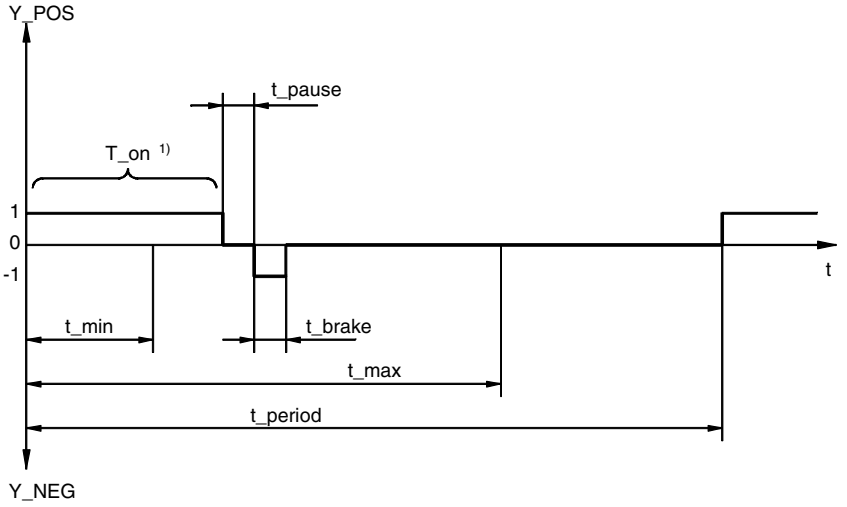
Der Parameter t_max gibt die maximale Impulslänge an, d.h. die Zeitdauer, die der Ausgang Y_POS bzw. der Ausgang Y_NEG maximal 1-Signal besitzen darf. Ist der nach obiger Gleichung berechnete Wert der Impulsdauer größer als t_max, so wird die Dauer des Impulses auf t_max begrenzt. Zwischen Stell- und Bremsimpuls muß zum Schutz der Leitungselektronik (Verhindern des gleichzeitigen Zündens antiparallel geschalteter Stromrichterventile) eventuell eine frei vorgebbare Pausenzeit von t_pause = 10 oder 20 ms eingelegt werden.

Der Parameter t_pause gibt die Zeitdauer an, die nach dem 1-Signal am Ausgang Y_POS (Y_NEG) gewartet werden soll, bevor für die Zeitdauer t_brake der entgegengesetzte Ausgang Y_NEG (Y_POS) auf 1-Signal gesetzt wird. Es handelt sich hierbei um einen Bremsimpuls der nach der Pausenzeit ausgegeben werden soll. Eine Pausenzeit von t_pause = 20 ms (t_pause = 0.02) entspricht einem Aussetzen der Zündwinkelsteuerung um zwei Halbwellen.

Damit ist ein genügend großer Sicherheitsabstand zur Verhinderung von Kurzschlüssen bzw. das Ansprechen der Schutzbeschaltung infolge des Zündens antiparalleler Ventile gewährleistet.

Darstellung der Zeitverhältnisse

Ein Verhältnis der Zeiten zueinander zeigt folgendes Bild:



1 Variable Einschaltzeit

Der Parameter up_pos kennzeichnet den Punkt der Eingangsgröße X, bei welcher der Ausgang Y_POS bei positiver Eingangsgröße X ständig 1-Signal liefern würde, wenn

t_pause = t_brake = 0

und

t_max = t_period.

Der Parameter up_neg kennzeichnet den Punkt der Eingangsgröße X, bei welcher der Ausgang Y_NEG bei negativer Eingangsgröße X ständig 1-Signal liefern würde, wenn

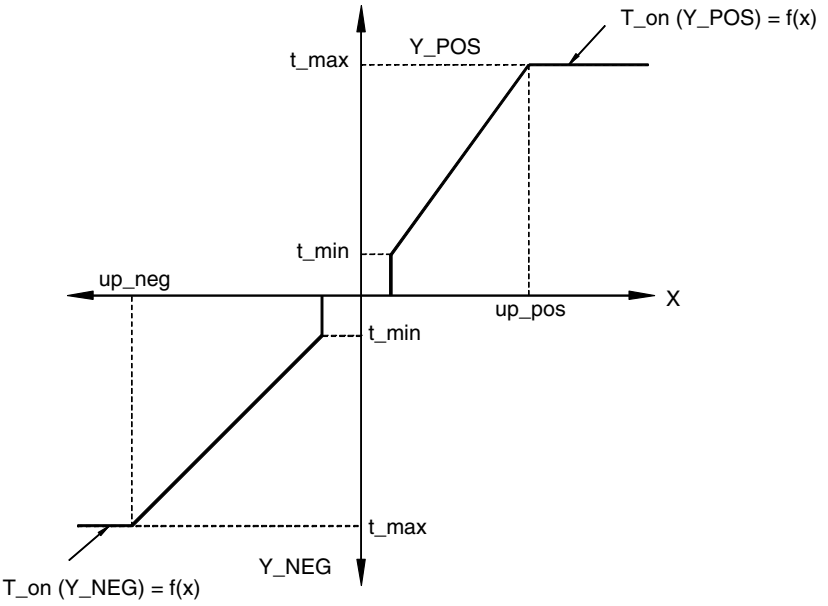
t_pause = t_brake = 0

und

t_max = t_period.

Abhängigkeit der Zeitdauer

Die Abhängigkeit der Zeitdauer, in welcher der Ausgang Y_POS (Y_NEG) 1-Signal hat, zu der Eingangsgröße X zeigt das folgende Bild (hier wird wiederum t_pause = t_brake = 0 angesetzt).



Betriebsarten

Bei der Betriebsart Reset R = 1 werden die Ausgänge Y_POS und Y_NEG auf "0" gesetzt. Die internen Zeitzähler werden ebenfalls normiert, so daß der Funktionsbaustein beim Übergang auf R = 0 mit der Ausgabe eines neuen 1-Signals am zugehörigen Ausgang beginnt.

Randbedingungen

Wird der Baustein PWM zusammen mit einem PID-Regler betrieben, so sollte die Periodendauer t_period so gewählt werden, daß sie der Abtastzeit des PID-Reglers entspricht. Dann ist nämlich gewährleistet, daß jedes neue Stellsignal vom PID-Regler innerhalb der Periodendauer vollständig bearbeitet werden kann.

Die Abtastzeit des PWM sollte danach gewählt werden, wie fein die Impulszeit zur Periodendauer steht. Durch sie wird der minimal mögliche Stellimpuls festgelegt

Es empfiehlt sich minimal folgendes Verhältnis:

t_period/Abtastzeit (PWM) ≥ 10

Beispiele zum Baustein PWM

Übersicht

In den Beispielen sind die Signalfolgen an den Ausgängen Y_POS und Y_NEG bei verschiedenen Werten des Eingangssignals X dargestellt. Die Beispiele unterscheiden sich hinsichtlich der gewählten Parametrierung.

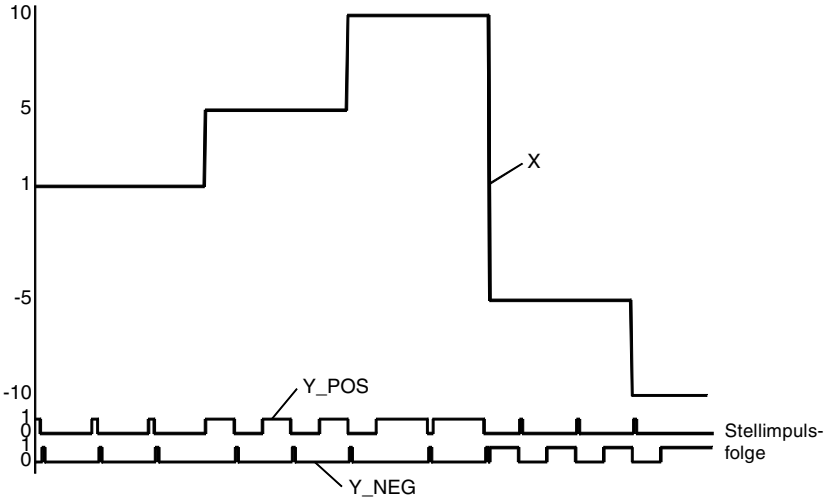
In diesem Abschnitt finden Sie folgende Beispiele zum Funktionsbaustein PWM

- Sprungantwort 1
- Sprungantwort 2

Sprungantwort 1 Für die Darstellung der Sprungantwort 1 gelten folgende Parametervorgaben:

Parameter	Vorgabe
t_period	4 s
t_min	0,2 s
t_max	3,8 s
t_pause	0,1 s
t_brake	0,2 s
up_pos	10
up_neg	10

Taktdiagramm Sprungantwort 1



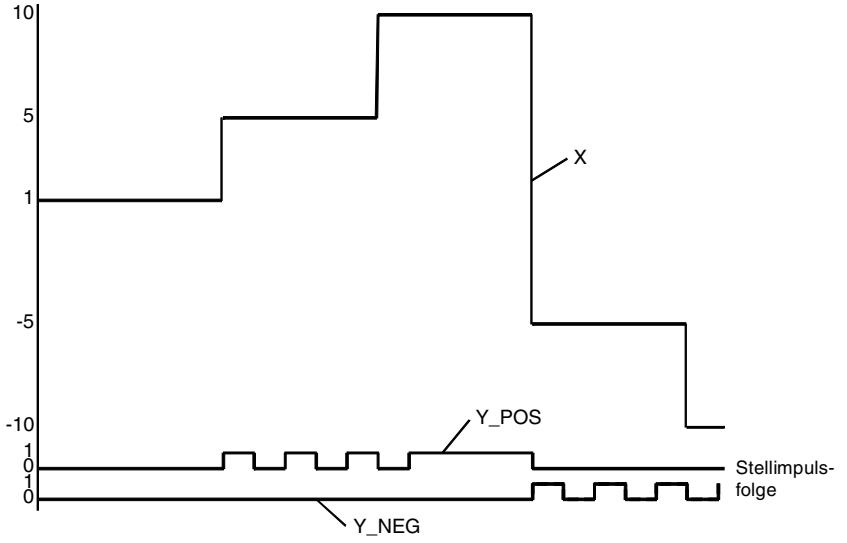
X analoges Signal

Deutlich zu erkennen ist, daß die Zeitdauer in der der Ausgang Y_POS 1-Signal führt mit größer werdendem Eingangssignal X auch größer wird. Ferner ist zu erkennen, daß nach jedem Y_POS-Signal ein kurzes Y_NEG-Signal (Ausgang Y_NEG auf 1-Signal) folgt. Dies ist auf die von "0" verschiedene Parametrierung von t_brake zurückzuführen. Bei negativem Eingangssignal X wird hier die Zeitdauer für das Signal Y_NEG länger, je negativer X wird. Auch hier folgt auf das Y_NEG-Signal ein kurzes Y_POS-Signal als Bremsimpuls.

Sprungantwort 2 Für die Darstellung der Sprungantwort 2 gelten folgende Parametervorgaben:

Parameter	Vorgabe
t_period	4 s
t_min	0,5 s
t_max	4 s
t_pause	0 s
t_brake	0 s
up_pos	10
up_neg	10

Taktdiagramm Sprungantwort 2



X analoges Signal

Der Unterschied zum Beispiel "Sprungantwort 1" ist, daß hier der Pause- und Bremsimpuls entfallen, da hier die entsprechenden Parameter zu "0" projektiert wurden. Ferner fällt auf, daß bei sehr kleinem Eingangssignal X kein Impuls mehr ausgegeben wird. Dies ist auf die Wirkung der Zeit t_min zurückzuführen. Außerdem wird bei großem Eingangssignal X (X = up_pos/up_neg) ein Dauerimpuls ausgegeben. Dies hängt damit zusammen, daß t_max = t_period gewählt wurde.

PWM1: Pulsbreitenmodulation

44

Übersicht

Einleitung

Dieses Kapitel beschreibt den Baustein PWM1.

Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Kurzbeschreibung	422
Darstellung	423
Formeln	424
Detailbeschreibung	425
Beispiel zum Baustein PWM1	427

Kurzbeschreibung

Verwendung des Bausteins

Die Ansteuerung von Stellgliedern erfolgt nicht nur über analoge Größen sondern auch mittels binärer Stellsignale.

Dabei soll die eingestellte mittlere Energie des Stellgliedes (Stellenergie) dem analogen Eingangswert (IN) des Modulationsbausteins entsprechen.

Funktionsbeschreibung

Der Funktionsbaustein PWM1 dient zur Umsetzung von analogen Werten in digitale Ausgangssignale für Concept.

Bei der Pulsbreitenmodulation (PWM1) wird bei konstanter Taktperiode ein in seiner Dauer vom Analogwert abhängiges 1-Signal ausgegeben. Die mittlere eingestellte Energie entspricht dem Quotienten aus der Einschaltdauer T_on und der Taktzeit t_period.

Damit die mittlere eingestellte Energie auch der analogen Eingangsgröße IN entspricht muß folgendes gelten:

$T_on \sim IN$

Als zusätzliche Parameter können EN und ENO projektiert werden.

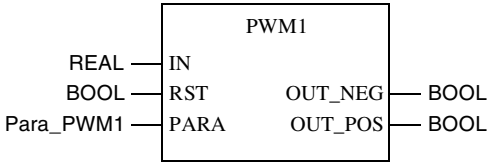
Allgemeines zur Stellgliedansteuerung

Im allgemeinen wird die binäre Stellgliedansteuerung durch zwei binäre Signale OUT_POS und OUT_NEG vorgenommen. Bei einem Motor entspricht der Ausgang OUT_POS beispielsweise dem Signal "Rechtslauf" und der Ausgang OUT_NEG dem Signal "Linkslauf". Bei einem Ofen können die Ausgänge OUT_POS und OUT_NEG entsprechend als "Heizen" und "Kühlen" interpretiert werden.

Darstellung

Symbol

Darstellung des Bausteins



Parameterbeschreibung PWM1

Beschreibung der Bausteinparameter

Parameter	Datentyp	Bedeutung
IN	REAL	Eingangsgröße
RST	BOOL	Betriebsart Reset ("1" = Reset)
PARA	Para_PWM1	Parameter
OUT_NEG	BOOL	Ausgang für negative IN-Werte
OUT_POS	BOOL	Ausgang für positive IN-Werte

Parameterbeschreibung Para_PWM1

Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
t_period	TIME	Periodendauer
t_min	TIME	Minimalzeit für Stellimpuls
in_max	REAL	Oberer Grenzwert für positive/negative IN-Werte

Formeln

Impulslänge für OUT_POS und OUT_NEG

Die Impulslänge T_on für die Ausgänge OUT_POS und OUT_NEG wird nach folgenden Gleichungen bestimmt:

Ausgang	Formel	Bedingung
OUT_POS	$T_on = t_period \times \frac{IN}{in_max}$	$0 \le IN \le in_max$
OUT_NEG	$T_on = t_period \times \frac{\lvert IN \rvert}{in_max}$	$0 \le -IN \le in_max$

Regeln zur Parametrierung

Für einen ordnungsgemäßen Betrieb ist bei der Parametrierung folgende Regel zu beachten:

$t_min \le t_period$

Detailbeschreibung

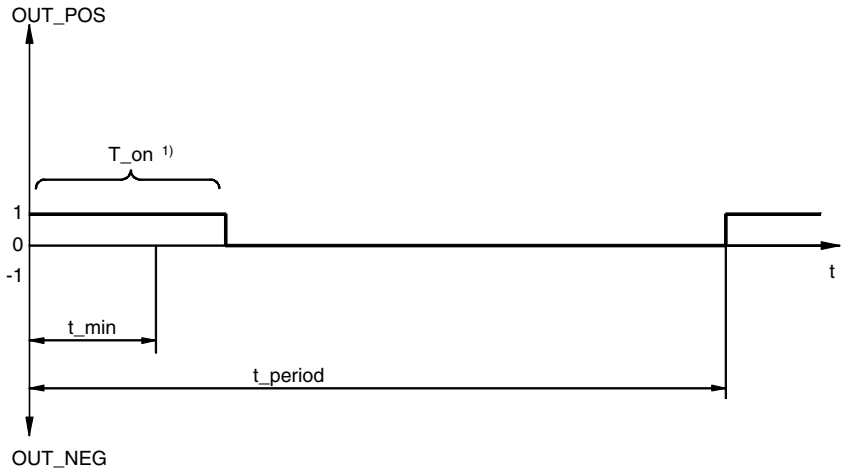
Arbeitsweise des Bausteins

Die Periodendauer legt die Zeit fest, in welcher die Stellimpulse (1-Signal am Ausgang OUT_POS bzw. OUT_NEG) regelmäßig d.h. im konstanten Zeitraster ausgegeben werden.

Der Parameter t_min gibt die minimale Impulslänge an, d.h. die Zeitdauer, die der Ausgang Y_POS bzw. der Ausgang Y_NEG minimal 1-Signal besitzen soll. Ist die nach der Gleichung im Abschnitt "*Formeln, S. 424*" berechnete Impulsdauer kleiner als t_min, so wird für die gesamte Periode kein Impuls ausgegeben.

Darstellung der Zeitverhältnisse

Ein Verhältnis der Zeiten zueinander zeigt folgendes Bild:

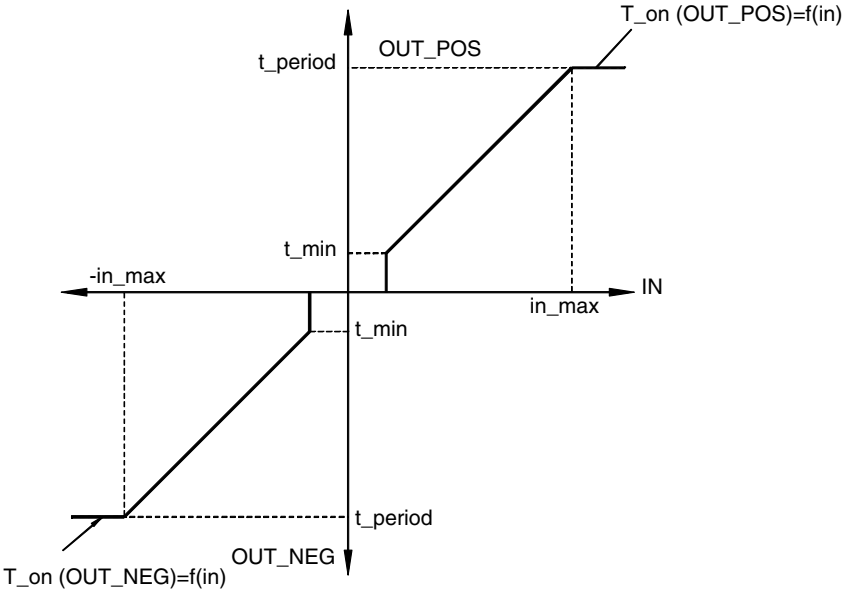


1 Variable Einschaltzeit

Der Parameter in_max kennzeichnet den Punkt der Eingangsgröße IN, bei welcher der Ausgang OUT_POS bei positiver Eingangsgröße IN ständig 1-Signal liefern würde.

Abhängigkeit der Zeitdauer

Die Abhängigkeit der Zeitdauer, in welcher der Ausgang OUT_POS (OUT_NEG) 1-Signal hat, zu der Eingangsgröße IN zeigt das folgende Bild:



Betriebsarten

Bei der Betriebsart Reset RST = 1 werden die Ausgänge OUT_POS und OUT_NEG auf "0" gesetzt. Die internen Zeitzähler werden ebenfalls normiert, so daß der Funktionsbaustein beim Übergang auf RST = 0 mit der Ausgabe eines neuen 1-Signals am zugehörigen Ausgang beginnt.

Rand-bedingungen

Wird der Baustein PWM1 zusammen mit einem PID-Regler betrieben, so sollte die Periodendauer t_period so gewählt werden, daß sie der Abtastzeit des PID-Reglers entspricht. Dann ist nämlich gewährleistet, daß jedes neue Stellsignal vom PID-Regler innerhalb der Periodendauer vollständig bearbeitet werden kann.

Die Abtastzeit des PWM1 sollte danach gewählt werden, wie fein die Impulszeit zur Periodendauer steht. Durch sie wird der minimal mögliche Stellimpuls festgelegt.

Es empfiehlt sich minimal folgendes Verhältnis:

t_period/Abtastzeit (PWM1) ≥ 10

Beispiel zum Baustein PWM1

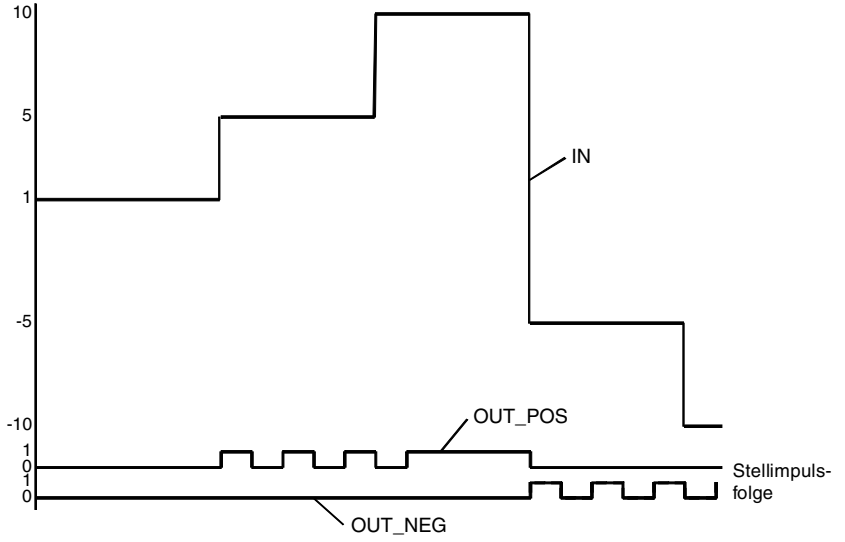
Sprungantwort

Im Beispiel sind die Signalfolgen an den Ausgängen OUT_POS und OUT_NEG bei verschiedenen Werten des Eingangssignals IN dargestellt.

Für die Darstellung der Sprungantwort gelten folgende Parametervorgaben:

Parameter	Vorgabe
t_period	4 s
t_min	0,5 s
in_max	10

Taktdiagramm Sprungantwort



IN Analoges Signal

Es fällt auf, daß bei sehr kleinem Eingangssignal IN kein Impuls mehr ausgegeben wird. Dies ist auf die Wirkung der Zeit t_min zurückzuführen. Bei einem großen Eingangssignal IN (IN = in_max) wird ein Dauerimpuls ausgegeben.

QDTIME: Totzeitglied

45

Übersicht

Einleitung

Dieses Kapitel beschreibt den Baustein QDTIME.

Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Kurzbeschreibung	430
Darstellung	431
Detailbeschreibung	432

Kurzbeschreibung

Funktionsbeschreibung

Mit diesem Funktionsbaustein wird ein Eingangssignal um eine Totzeit verzögert.

Der Funktionsbaustein verzögert das Signal IN um die Totzeit T_DELAY, bevor es an OUT wieder ausgegeben wird.

Der Funktionsbaustein hat einen Delay-Puffer für 128 Elemente (IN-Werte), d.h. während der Zeit T_DELAY können 128 IN-Werte gespeichert werden. Der Puffer wird entsprechend den verschiedenen Betriebsarten behandelt.

Nach einem Kalt- oder Warmstart des Systems bleibt der Wert von OUT unverändert. Die internen Werte werden auf den Wert von IN gesetzt.

Nach einem Kalt- oder Warmstart des Systems oder einer Änderung der Totzeit T_DELAY, wird der Ausgang READY gleich "0". Dies bedeutet: nicht bereit, leerer Puffer.

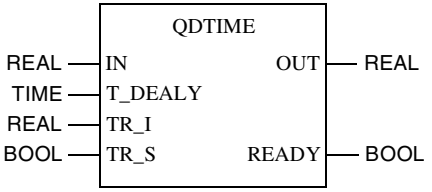
Der Funktionsbaustein hat die Betriebsarten Tracking und Automatik.

Als zusätzliche Parameter können EN und ENO projektiert werden.

Darstellung

Symbol

Darstellung des Bausteins



Parameterbeschreibung

Beschreibung der Bausteinparameter

Parameter	Datentyp	Bedeutung
IN	REAL	Eingangswert
T_DELAY	TIME	Totzeit
TR_I	REAL	Initialisierungs-Eingang
TR_S	BOOL	Initialisierungs-Typ: "1" = Betriebsart Tracking "0" = Betriebsart Automatik
OUT	REAL	Ausgang
READY	BOOL	"1" = Interner Puffer gefüllt "0" = Interner Puffer nicht gefüllt (z.B. nach Warm-/ Kaltstart oder Änderung der Totzeit)

Detailbeschreibung

Auswahl der Betriebsarten

Es gibt zwei Betriebsarten, die über den Eingang TR_S ausgewählt werden:

Betriebsart	TR_S
Automatik	0
Tracking	1

Betriebsart Automatik

In der Betriebsart Automatik arbeitet der Funktionsbaustein nach folgenden Regeln:

Wenn...	Dann...
Zykluszeit > T_DELAY/128	wird der aktuelle IN-Wert in den Puffer übertragen, und der älteste IN-Wert im Puffer wird am Ausgang OUT ausgegeben. In diesem Fall ist die Auflösung kleiner 128, und es kommt zu einem systematischen Fehler, d.h. einige IN-Werte werden doppelt gespeichert (siehe auch Beispiel).
Zykluszeit < T_DELAY/128	können nicht alle IN-Werte im Puffer abgelegt werden. In diesem Fall wird in einigen Zyklen der IN-Wert nicht gespeichert, und OUT bleibt in diesem Zyklen unverändert

Beispiel Zykluszeit > 128

Folgende Werte werden angenommen:

Zykluszeit = 100 ms

T_DELAY = 10 s

tin = T_DELAY / 128 = 78 ms

Da tin (Einlesezeit) kleiner ist als die Zykluszeit, wird jeder IN-Wert in den Puffer übernommen. Bei der vierten Ausführung des Funktionsbausteins (nach 400 ms) wird der IN-Wert jedoch zweimal anstatt einmal gespeichert (weil 3 x 78 = 312 und 4 x 78 = 390).

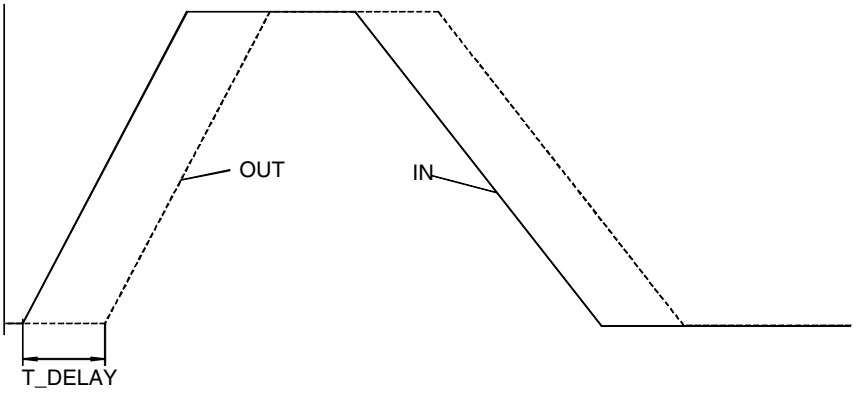
Betriebsart Tracking

In der Betriebsart Tracking wird der Trackingwert TR_I fest auf den Ausgang OUT übertragen. Der interne Puffer wird mit dem Trackingwert TR_I gefüllt. Der Puffer wird als gefüllt markiert (READY = 1).

Beispiel zum Verhalten des QDTIME

Das Diagramm zeigt ein Beispiel für das Verhalten des Funktionsbausteins. Der Eingang IN ändert sich in Form einer Rampe von einem Wert auf einen neuen Wert und der Ausgang OUT folgt um die Totzeit T_DELAY verzögert dem Eingang IN.

Diagramm Funktionsbaustein QDTIME



QPWM: Pulsbreitenmodulation (einfach)

46

Übersicht

Einleitung

Dieses Kapitel beschreibt den Baustein QPWM.

Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Kurzbeschreibung	436
Darstellung	437
Formeln	438
Detailbeschreibung	439
Beispiel zum Baustein QPWM	441

Kurzbeschreibung

Verwendung des Bausteins

Die Ansteuerung von Stellgliedern erfolgt nicht nur über analoge Größen sondern auch mittels binärer Stellsignale. Die Wandlung von analogen Werten in binäre Ausgangssignale erfolgt beispielsweise über Pulsbreitenmodulation (QPWM) oder Pulsdauermodulation (PDM).

Dabei soll die eingestellte mittlere Energie des Stellgliedes (Stellenergie) dem analogen Eingangswert (X) des Modulationsbausteins entsprechen.

Funktionsbeschreibung

Der Funktionsbaustein QPWM dient zur Umsetzung von analogen Werten in digitale Ausgangssignale.

Bei der Pulsbreitenmodulation (QPWM) wird bei konstanter Taktperiode ein in seiner Dauer vom Analogwert abhängiges 1-Signal ausgegeben. Die mittlere eingestellte Energie entspricht dem Quotienten aus der Einschaltdauer T_on und der Taktzeit t_period.

Damit die mittlere eingestellte Energie auch der analogen Eingangsgröße X entspricht muß folgendes gelten:

$T_on \sim X$

Als zusätzliche Parameter können EN und ENO projektiert werden.

Allgemeines zur Stellgliedansteuerung

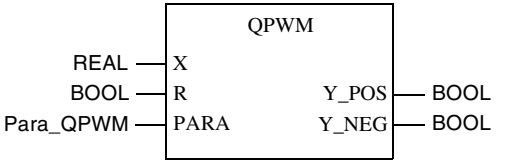
Im allgemeinen wird die binäre Stellgliedansteuerung durch zwei binäre Signale Y_POS und Y_NEG vorgenommen.

Bei einem Motor entspricht der Ausgang Y_POS beispielsweise dem Signal "Rechtslauf" und der Ausgang Y_NEG dem Signal "Linkslauf". Bei einem Ofen können die Ausgänge Y_POS und Y_NEG entsprechend als "Heizen" und "Kühlen" interpretiert werden.

Darstellung

Symbol

Darstellung des Bausteins



Parameterbeschreibung QPWM

Beschreibung der Bausteinparameter

Parameter	Datentyp	Bedeutung
X	REAL	Eingangsgröße
R	BOOL	Betriebsart Reset ("1" = Reset)
PARA	Para_QPWM	Parameter
Y_POS	BOOL	Ausgang für positive X-Werte
Y_NEG	BOOL	Ausgang für negative X-Werte

Parameterbeschreibung Para_QPWM

Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
t_period	TIME	Periodendauer
t_min	TIME	Minimalzeit für Stellimpuls
x_max	REAL	Oberer Grenzwert für positive/negative X-Werte

Formeln

Impulslänge für Y_POS und Y_NEG

Die Impulslänge T_on für die Ausgänge Y_POS und Y_NEG wird nach folgenden Gleichungen bestimmt:

Ausgang	Formel	Bedingung
Y_POS	$T_on = t_period \times \frac{X}{x_max}$	$0 \le X \le x_max$
Y_NEG	$T_on = t_period \times \frac{\lvert X \rvert}{x_max}$	$0 \le -X \le x_max$

Regeln zur Parametrierung

Für einen ordnungsgemäßen Betrieb ist bei der Parametrierung folgende Regel zu beachten:

$t_min \le t_period$

Detailbeschreibung

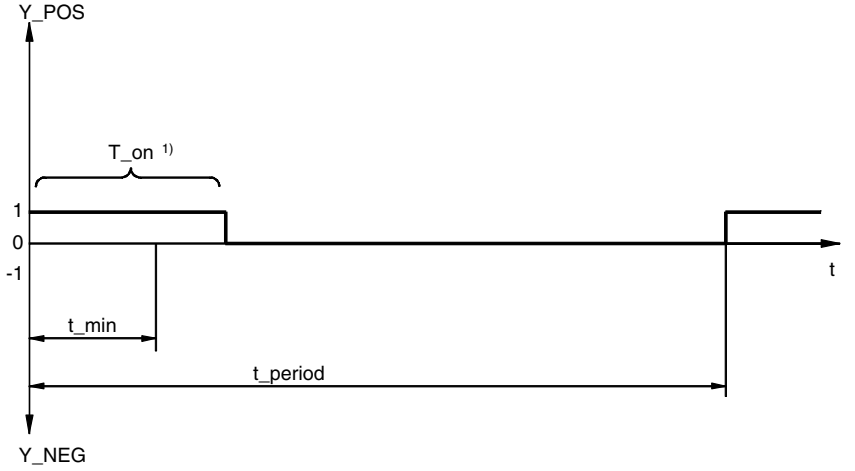
Arbeitsweise des Bausteins

Die Periodendauer legt die Zeit fest, in welcher die Stellimpulse (1-Signal am Ausgang Y_POS bzw. Y_NEG) regelmäßig d.h. im konstanten Zeitraster ausgegeben werden.

Der Parameter t_min gibt die minimale Impulslänge an, d.h. die Zeitdauer, die der Ausgang Y_POS bzw. der Ausgang Y_NEG minimal 1-Signal besitzen soll. Ist die nach der Gleichung im Abschnitt "*Formeln, S. 438*" berechnete Impulsdauer kleiner als t_min, so wird für die gesamte Periode kein Impuls ausgegeben.

Darstellung der Zeitverhältnisse

Ein Verhältnis der Zeiten zueinander zeigt folgendes Bild:

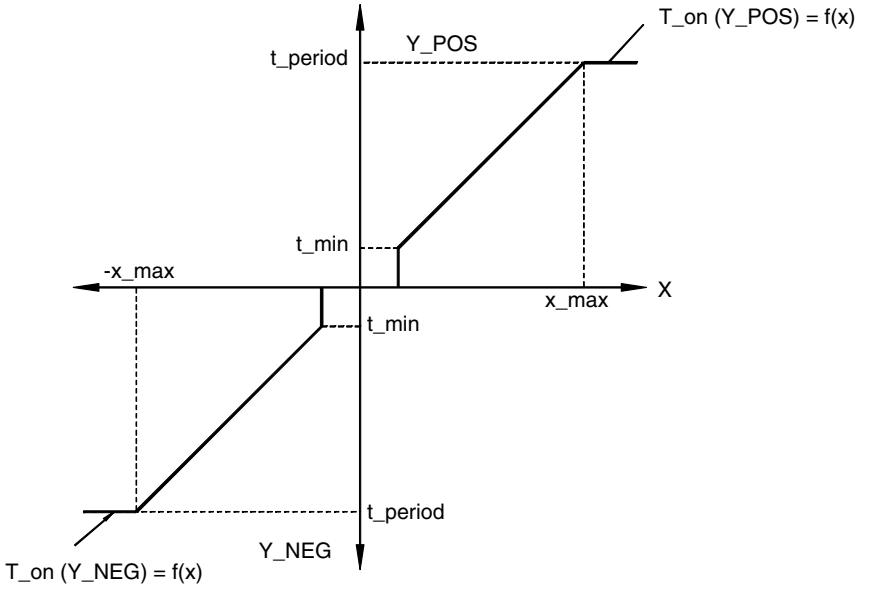


1 Variable Einschaltzeit

Der Parameter x_max kennzeichnet den Punkt der Eingangsgröße X, bei welcher der Ausgang Y_POS bei positiver Eingangsgröße X ständig 1-Signal liefern würde.

Abhängigkeit der Zeitdauer

Die Abhängigkeit der Zeitdauer, in welcher der Ausgang Y_POS (Y_NEG) 1-Signal hat, zu der Eingangsgröße X zeigt das folgende Bild :



Betriebsarten

Bei der Betriebsart Reset R = 1 werden die Ausgänge Y_POS und Y_NEG auf "0" gesetzt. Die internen Zeitzähler werden ebenfalls normiert, so daß der Funktionsbaustein beim Übergang auf R = 0 mit der Ausgabe eines neuen 1-Signals am zugehörigen Ausgang beginnt.

Randbedingungen

Wird der Baustein QPWM zusammen mit einem PID-Regler betrieben, so sollte die Periodendauer t_period so gewählt werden, daß sie der Abtastzeit des PID-Reglers entspricht. Dann ist nämlich gewährleistet, daß jedes neue Stellsignal vom PID-Regler innerhalb der Periodendauer vollständig bearbeitet werden kann.

Die Abtastzeit des QPWM sollte danach gewählt werden, wie fein die Impulszeit zur Periodendauer steht. Durch sie wird der minimal mögliche Stellimpuls festgelegt.

Es empfiehlt sich minimal folgendes Verhältnis:

t_period/Abtastzeit (QPWM) ≥ 10

Beispiel zum Baustein QPWM

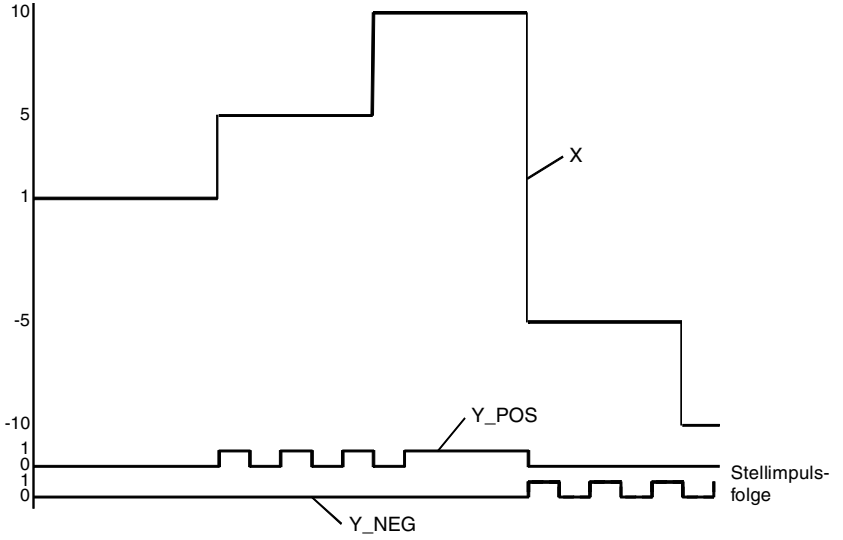
Sprungantwort

Im Beispiel sind die Signalfolgen an den Ausgängen Y_POS und Y_NEG bei verschiedenen Werten des Eingangssignals X dargestellt.

Für die Darstellung der Sprungantwort gelten folgende Parametervorgaben:

Parameter	Vorgabe
t_period	4 s
t_min	0,5 s
x_max	10

Taktdiagramm Sprungantwort



X Analoges Signal

Es fällt auf, daß bei sehr kleinem Eingangssignal X kein Impuls mehr ausgegeben wird. Dies ist auf die Wirkung der Zeit t_min zurückzuführen. Bei einem großem Eingangssignal X (X = x_max) wird ein Dauerimpuls ausgegeben.

RAMP: Rampengenerator

47

Übersicht

Einleitung

Dieses Kapitel beschreibt den Baustein RAMP.

Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Kurzbeschreibung	444
Darstellung	445
Detailbeschreibung	446
Laufzeitfehler	448

Kurzbeschreibung

Funktionsbeschreibung

Der Funktionsbaustein RAMP ermöglicht es, von einem anfänglichen Sollwert rampenförmig auf einen Zielwert zu fahren. Die Steigungen der positiven und negativen Rampen können unterschiedlich sein.

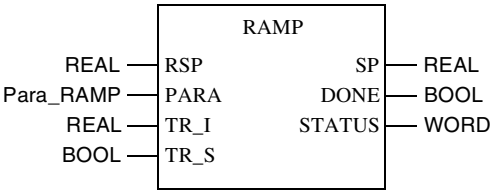
Eine Anzeige (DONE-Ausgang) informiert den Anwender, ob der Zielwert bereits erreicht ist oder ob die Rampe gerade durchgeführt wird.

Als zusätzliche Parameter können EN und ENO projektiert werden.

Darstellung

Symbol

Darstellung des Bausteins



Parameterbeschreibung RAMP

Beschreibung der Bausteinparameter

Parameter	Datentyp	Bedeutung
RSP	REAL	Zielwert der Rampe
PARA	Para_RAMP	Parameter
TR_I	REAL	Initialwert der Rampe
TR_S	BOOL	Initialisierungsbefehl der Rampe
SP	REAL	Ausgang
DONE	BOOL	"1": der Zielwert wurde erreicht "0": Rampenfunktion läuft ab
STATUS	WORD	Statuswort

Parameterbeschreibung Para_RAMP

Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
inc_rate	REAL	Positive Steigung in Einheiten pro Sekunde (≥0)
dec_rate	REAL	Negative Steigung in Einheiten pro Sekunde (≥0)

Detailbeschreibung

Parametrierung

Wenn der am Eingang vorgegebene Wert (RSP) den aktuellen Wert des SP-Ausgangs übersteigt, erhöht der Funktionsbaustein den Wert dieses Ausgangs mit der Geschwindigkeit inc_rate solange, bis der SP-Wert den RSP-Wert erreicht. Wenn inc_rate Null ist, wird die Rampenfunktion nicht ausgeführt, SP ist die direkte Kopie von RSP.

Wenn der am Eingang vorgegebene Wert den aktuellen Wert von SP unterschreitet, senkt der Funktionsbaustein den Wert von SP mit der Geschwindigkeit dec_rate. Wenn dec_rate Null ist, wird die Rampenfunktion nicht durchgeführt, SP ist die direkte Kopie von RSP.

Wenn der Wert von RSP sich während der Generierung der Rampe ändert, versucht der Funktionsbaustein unverzüglich, diesen neuen Zielwert zu erreichen. Die gerade ablaufende Rampenfunktion wird weitergeführt oder ändert ihre Richtung.

Betriebsarten

Der Tracking-Betrieb (TR_S = 1) ermöglicht es, dem SP-Ausgang einen Initialwert zuzuweisen. Sie gehen wie folgt vor:

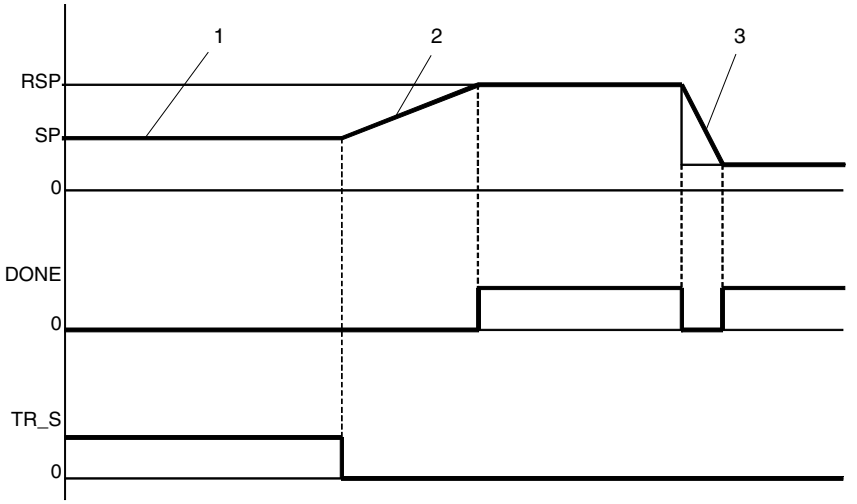
Schritt	Aktion
1	TR_I auf den gewünschten Initialwert setzen.
2	TR_S auf 1 setzen, der TR_I-Eingang wird in diesem Fall kontinuierlich auf SP geführt. **Hinweis:** Im Trackingbetrieb (TR_S = 1) verbleibt der DONE-Ausgang permanent auf Null.
3	Wenn TR_S auf 0 gesetzt wird, nimmt der Funktionsbaustein wieder seinen normalen Betrieb auf: SP nähert sich kontinuierlich RSP, wobei der Wert eine Rampe beschreibt.

DONE-Anzeige

Der DONE-Ausgang geht auf 1 über, wenn gerade eine Rampenfunktion beendet wurde. Er wird wieder auf Null gesetzt, wenn eine neue Rampe beginnt oder wenn der Funktionsbaustein auf Trackingbetrieb umgestellt wird.

Taktdiagramm

Taktdiagramm zum Baustein RAMP



1 Initialisierung: SP = TR_I
2 Steigende Rampe = inc_rate
3 Fallende Rampe = dec_rate

Laufzeitfehler

Statuswort

Im Statuswort werden folgende Meldungen angezeigt:

Bit	Bedeutung
Bit 0 = 1	Fehler in einer Berechnung mit Gleitkommawerten
Bit 1= 1	Erfassung eines unzulässigen Wertes an einem der Gleitkommawerteingänge
Bit 2= 1	Division durch Null bei einer Berechnung mit Gleitkommawerten
Bit 3 = 1	Kapazitätsüberlauf bei einer Berechnung mit Gleitkommawerten
Bit 4 = 1	Eine der folgenden Größen ist negativ : inc_rate, dec_rate. Zur Berechnung verwendet der Funktionsbaustein den Wert 0.

Fehlermeldung

Ein Fehler wird gemeldet, wenn an einem Eingang ein nicht gleitender Wert erfaßt wird oder wenn in einer Berechnung mit Gleitkommawerten ein Problem auftritt. In diesem Fall bleiben die Ausgänge SP und DONE unverändert.

Warnung

In folgenden Fällen erscheint eine Warnung:

- Der Parameter inc_rate ist negativ: der Funktionsbaustein verwendet den Wert 0 anstelle des fehlerhaften Wertes von inc_rate.
- Der Parameter dec_rate ist negativ: der Funktionsbaustein verwendet den Wert 0 anstelle des fehlerhaften Wertes von dec_rate.

RATIO: Verhältnisregler

48

Übersicht

Einleitung

Dieses Kapitel beschreibt den Baustein RATIO.

Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Kurzbeschreibung	450
Darstellung	451
Detailbeschreibung	452
Laufzeitfehler	454

Kurzbeschreibung

Funktionsbeschreibung

Der Funktionsbaustein RATIO führt, wenn er an einen Regler angeschlossen wird, eine Verhältnisregelung durch.

Eine Verhältnisregelung hat zum Ziel, einen Istwert PV (Regelgröße) zu einem anderen Istwert PV_TRACK (Führungsgröße) in Beziehung zu setzen. Die Rolle des Funktionsbausteins RATIO besteht darin, den Reglersollwert entsprechend der Steuergröße zu berechnen.

Als zusätzliche Parameter können EN und ENO projektiert werden.

Eigenschaften

Der Funktionsbaustein besitzt folgende Eigenschaften:

- Das Verhältnis kann intern (K) oder extern gesteuert werden (RK)
- Oberer und unterer Grenzwert für K oder RK
- Oberer und unterer Grenzwert für den berechneten Sollwert SP
- Berechnung des realen Verhältnisses: KACT = (PV - bias) / PV_TRACK

Formel

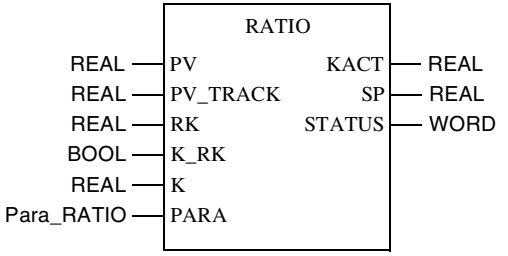
Berechnung des Reglersollwertes

$$SP = K \times PV_TRACK + bias$$

Darstellung

Symbol

Darstellung des Bausteins



Parameterbeschreibung RATIO

Beschreibung der Bausteinparameter

Parameter	Datentyp	Bedeutung
PV	REAL	Vom Regelkreis geregelter Istwert (wird nur zur Berechnung von KACT verwendet)
PV_TRACK	REAL	Führungsgröße des Regelkreises
RK	REAL	Externer Verhältniskoeffizient
K_RK	BOOL	Koeffiziententyp für verwendetes Verhältnis "1": externes Verhältnis RK "0": internes Verhältnis K
K	REAL	Koeffizient für internes Verhältnis
PARA	Para_RATIO	Parameter
KACT	REAL	Koeffizient für reales Verhältnis
SP	REAL	Berechneter Ausgang
STATUS	WORD	Statuswort

Parameterbeschreibung Para_RATIO

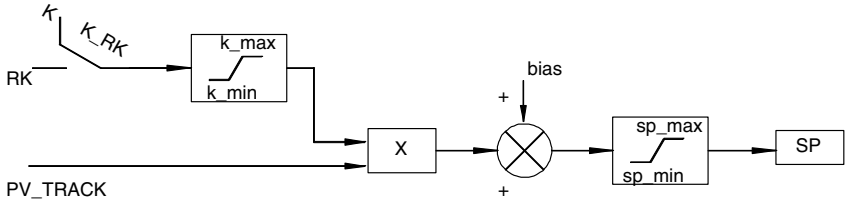
Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
k_min	REAL	Unterer Grenzwert bei Verhältnis K oder RK
k_max	REAL	Oberer Grenzwert bei Verhältnis K oder RK
sp_min	REAL	Unterer Grenzwert des berechneten Ausgangs SP
sp_max	REAL	Oberer Grenzwert des berechneten Ausgangs SP
bias	REAL	Offset-Koeffizient

Detailbeschreibung

Strukturbild

Strukturbild des Bausteins RATIO

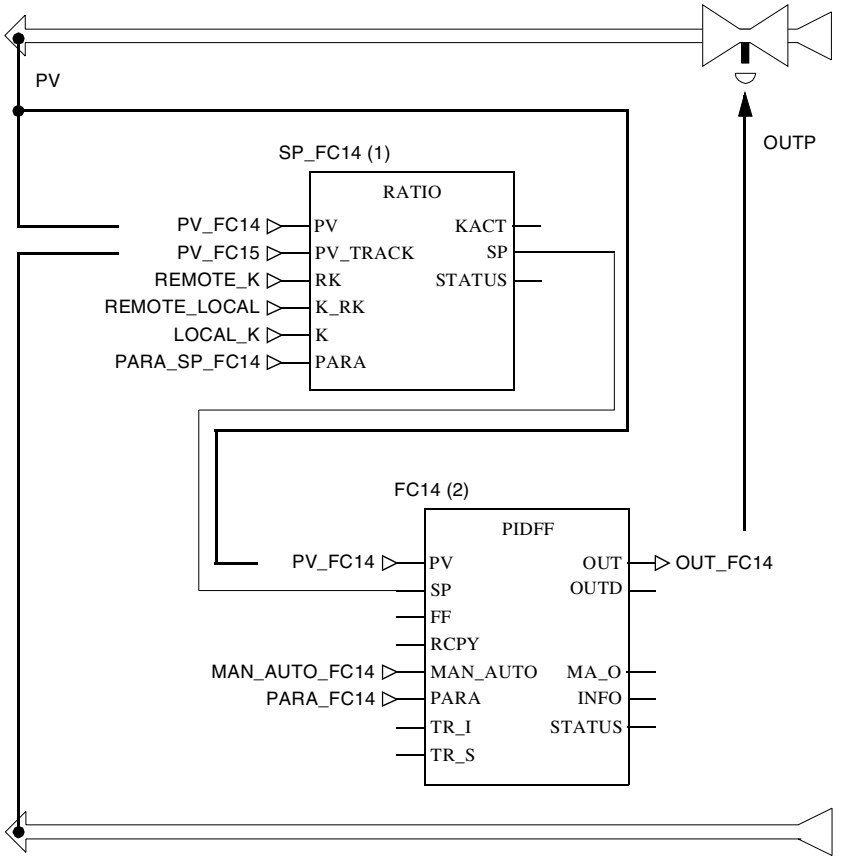


Anwendung

Der Funktionsbaustein RATIO ist einem Verhältnisregler vorgeschaltet. Seine Aufgabe besteht darin, den externen Sollwert SP eines nachfolgend aufgeschalteten Reglers zu berechnen. Der Verhältnisregler muß bestehen aus den Funktionsbausteinen RATIO, SP_SEL sowie einem Regler.

Typischerweise wird dieser Reglertyp zur Regelung eines Durchflusses im Verhältnis zu einem anderen gemessenen Durchfluß eingesetzt, wobei er ein bestimmtes Verhältnis K zwischen den beiden Durchflußmengen einhält.

Darstellung des Verhältnisreglers



Laufzeitfehler

Statuswort

Im Statuswort werden folgende Meldungen angezeigt:

Bit	Bedeutung
Bit 0 = 1	Fehler in einer Berechnung mit Gleitkommawerten
Bit 1= 1	Erfassung eines unzulässigen Wertes an einem der Gleitkommawerteingänge
Bit 2= 1	Division durch Null bei einer Berechnung mit Gleitkommawerten
Bit 3 = 1	Kapazitätsüberlauf bei einer Berechnung mit Gleitkommawerten
Bit 4 = 1	Der Eingang K (oder RK) liegt außerhalb des Bereichs [k_min, k_max]: Zur Berechnung verwendet der Funktionsbaustein den Wert k_min oder k_max.
Bit 5 = 1	Der Ausgang SP hat den unteren Grenzwert sp_min erreicht: SP wird auf sp_min begrenzt
Bit 6 = 1	Der Ausgang SP hat den oberen Grenzwert sp_max erreicht: SP wird auf sp_max begrenzt

Fehlermeldung

Der Fehler erscheint, wenn an einem Eingang ein nicht gleitender Wert erfaßt wird oder wenn bei einer Berechnung mit Gleitkommawerten ein Problem auftaucht. Die Ausgänge KACT und SP bleiben unverändert.

SCALING: Skalierung

49

Übersicht

Einleitung

Dieses Kapitel beschreibt den Baustein SCALING.

Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Kurzbeschreibung	456
Darstellung	456
Parametrierung	457
Laufzeitfehler	458

Kurzbeschreibung

Funktionsbeschreibung

Dieser Funktionsbaustein ermöglicht es, eine numerische Größe in ihrer Größe zu verändern.

Als zusätzliche Parameter können EN und ENO projektiert werden.

Formel

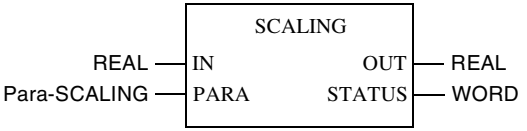
Der Funktionsbaustein führt folgende Berechnung durch:

$$OUT = (IN - in_min) \times \frac{(out_max - out_min)}{(in_max - in_min)} + out_min$$

Darstellung

Symbol

Darstellung des Bausteins



Parameterbeschreibung SCALING

Beschreibung der Bausteinparameter

Parameter	Datentyp	Bedeutung
IN	REAL	Zu skalierende numerische Größe
PARA	Para_SCALING	Parameter
OUT	REAL	Skalierter Ausgangswert
STATUS	WORD	Statuswort

Parameterbeschreibung Para_SCALING

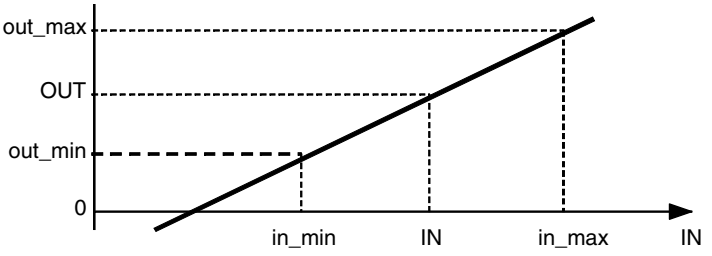
Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
in_min	REAL	Unterer Grenzwert der Eingangsskala
in_max	REAL	Oberer Grenzwert der Eingangsskala.
out_min	REAL	Unterer Grenzwert der Ausgangsskala
out_max	REAL	Oberer Grenzwert der Ausgangsskala
clip	BOOL	"1": der Wert des Ausgangangs OUT ist begrenzt zwischen out_min und out_max.

Parametrierung

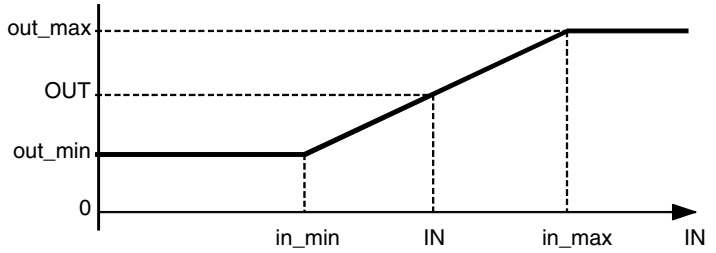
Ohne Ausgangsbegrenzung (clip = 0)

Wenn der Clip-Parameter auf 0 eingestellt ist, ist die Skalierung unabhängig vom Wert des Eingangs IN.



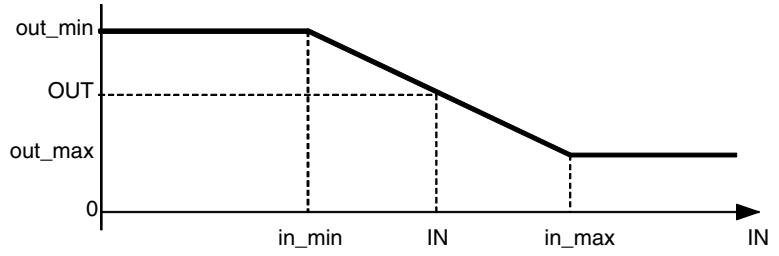
Mit Ausgangsbegrenzung (clip =1)

Wenn der Clip-Parameter auf 1 eingestellt ist, erfolgt die Skalierung innerhalb des Bereichs [in_min , in_max]. Außerhalb dieses Bereichs wird der Ausgang auf die Werte out_min und out_max begrenzt.



Änderung der Anstiegsrichtung

Es ist ebenfalls möglich, die Anstiegsrichtung der numerischen Eingangsgröße zu ändern, indem out_max ein geringerer Wert zugeordnet wird als out_min.



Laufzeitfehler

Statuswort

Im Statuswort werden folgende Meldungen angezeigt:

Bit	Bedeutung
Bit 0 = 1	Fehler in einer Berechnung mit Gleitkommawerten
Bit 1= 1	Erfassung eines unzulässigen Wertes an einem der Gleitkommawerteingänge
Bit 2= 1	Division durch Null bei einer Berechnung mit Gleitkommawerten
Bit 3 = 1	Kapazitätsüberlauf bei einer Berechnung mit Gleitkommawerten
Bit 4 = 1	Der Clip-Parameter ist auf 1 eingestellt und der Eingang IN liegt außerhalb des Bereichs [in_min, in_max]: zur Berechnung benötigt der Funktionsbaustein den Wert in_min bzw. in_max.
Bit 7 = 1	Der Parameter in_min ist gleich in_max

Fehlermeldung

Ein Fehler wird in folgenden Fällen gemeldet:

- An einem Eingang wird ein nicht gleitender Wert erfaßt.
- Bei einer Berechnung mit Gleitkommawerten tritt ein Problem auf.
- Wenn in_min = in_max

In diesen Fällen bleibt der Ausgang OUT unverändert.

SCON3: Dreipunkt-Schrittregler

50

Übersicht

Einleitung

Dieses Kapitel beschreibt den Baustein SCON3.

Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Kurzbeschreibung	460
Darstellung	461
Detailbeschreibung	462
Laufzeitfehler	464

Kurzbeschreibung

Funktionsbeschreibung

Der Funktionsbaustein bildet einen Dreipunkt-Schrittregler nach, der durch eine dynamische Rückführung PD-ähnliches Verhalten erhält.

Als zusätzliche Parameter können EN und ENO projektiert werden.

Eigenschaften

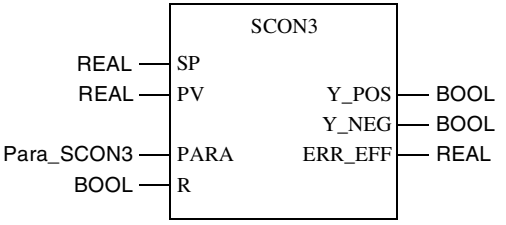
Der Funktionsbaustein SCON3 besitzt folgende Eigenschaften:

- Betriebsarten Reset, Automatik
- eine interne Rückführung (Verzögerer 1. Ordnung)

Darstellung

Symbol

Darstellung des Bausteins



Parameterbeschreibung SCON3

Beschreibung der Bausteinparameter

Parameter	Datentyp	Bedeutung
SP	REAL	Sollwerteingang
PV	REAL	Istwerteingang
PARA	Para_SCON3	Parameter
R	BOOL	"1" = Betriebsart Reset
ERR_EFF	REAL	Wirksamer Schaltwert
Y_POS	BOOL	"1" = positive Stellgröße an Ausgang ERR_EFF
Y_NEG	BOOL	"1" = negative Stellgröße an Ausgang ERR_EFF

Parameterbeschreibung Para_SCON3

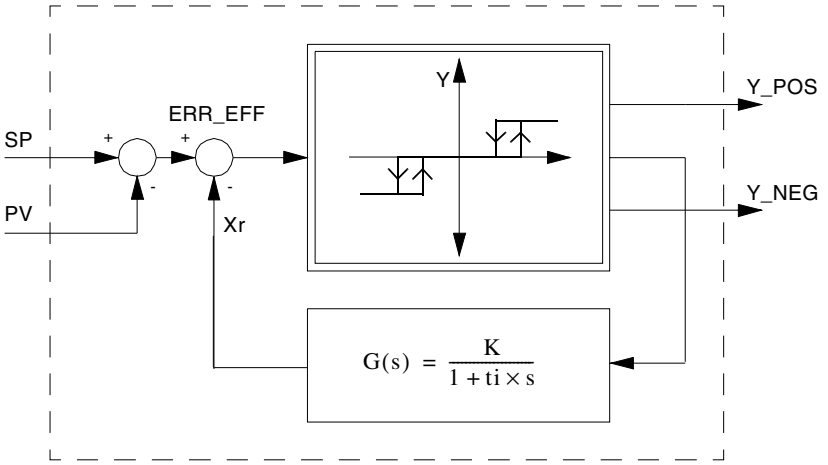
Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
gain	REAL	Proportionalbeiwert (Verstärkungsfaktor)
ti	TIME	Nachstellzeit
t_proc	TIME	Nennstellzeit des angeschlossenen Ventils
hys	REAL	Hysterese vom Dreipunktschalter
db	REAL	Unempfindlichkeitszone

Detailbeschreibung

Struktur des Reglers

Struktur des Dreipunktreglers:



Abhängigkeit der Ausgänge Y_POS und Y_NEG von der Größe Y:

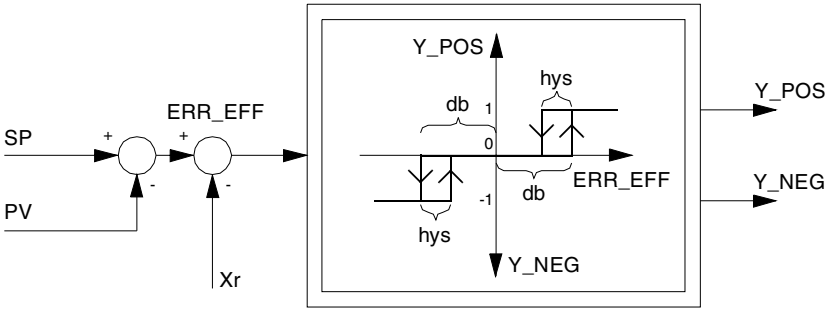
Wenn...	Dann...
Y = 1	Y_POS = 1 Y_NEG = 0
Y = 0	Y_POS = 0 Y_NEG = 0
Y = -1	Y_POS = 0 Y_NEG = 1

Bedeutung der Größe K:

$$K = \frac{ti}{t_proc \times gain}$$

Prinzip des Dreipunktreglers

Dem eigentlichen Dreipunkt-Regler wird eine dynamische Rückführung (PT1-Glied hinzugefügt. Durch geeignete Wahl der Zeitkonstante ti und t_proc dieses Rückführgliedes erhält der Dreipunkt-Regler ein dynamisches Verhalten, daß dem Verhalten eines PD-Reglers entspricht.



Die Verstärkung gain ist größer 0 zu wählen.

Unempfindlichkeitszone

Der Parameter db legt den Einschaltpunkt für die Ausgänge Y_POS und Y_NEG fest. Ist der wirksame Schaltwert ERR_EFF = SP - PV- XR positiv und wird größer als db, dann schaltet der Ausgang Y_POS von "0" auf "1". Ist der wirksame Schaltwert ERR_EFF negativ und wird kleiner als -db, dann schaltet der Ausgang Y_NEG von "0" auf "1". Der Wert des Parameters db wird typisch auf 1% des maximalen Regelbereiches (max. (SP - PV)) eingestellt.

Hinweis: Von der Unempfindlichkeitszone db wird der Betrag ausgewertet!

Hysterese

Der Parameter hys gibt die Schalthysterese an, d.h. den Wert, um den sich der wirksame Schaltwert ERR_EFF ausgehend vom Schaltpunkt db verkleinern muß bevor der Ausgang Y_POS (Y_NEG) wieder auf "0" gesetzt wird. Der Zusammenhang zwischen Y_POS und Y_NEG abhängig von dem wirksamen Schaltwert ERR_EFF und den Parametern db und hys wird im Bild "*Prinzip des Dreipunktreglers, S. 463*" deutlich. Der Wert des Parameters hys wird typisch auf 0.5 % des maximalen Regelbereiches (max. (SP - PV)) eingestellt.

Hinweis: Von der Hysterese hys wird der Betrag ausgewertet!

Verhalten bei fehlerhafter Zeitkonstante

Ist die Zeitkonstante ti = 0 oder die Verstärkung gain ≤ 0 (Fehler in der Projektierung), so wird der Baustein dennoch bearbeitet. Allerdings wird die Funktion der Rückführung abgeschaltet, so daß der Baustein wie ein gewöhnlicher Dreipunktschalter arbeitet.

Ist die Zeitkonstante t_proc = 0 (Fehler in der Projektierung), so wird der Baustein dennoch bearbeitet. In diesem Fall wird für t_proc ein voreingestellter Wert von t_proc = 60 s (60 000 msec) eingestellt.

Betriebsarten

Es gibt zwei Betriebsarten, die über den Eingang R ausgewählt werden:

Betriebsart	R	Bedeutung
Automatik	0	Der Funktionsbaustein wird, wie vor beschrieben, bearbeitet.
Reset	1	Der interne Wert des Rückführgliedes wird auf SP - PV gesetzt. Die Ausgänge Y_POS und Y_NEG werden beide auf "0" gesetzt.

Laufzeitfehler

Fehlermeldung

Bei hys > 2 * db erfolgt eine Fehlermeldung.

Warnung

In folgenden Fällen erfolgt eine Warnung:

- gain ≤ 0 : Der Regler arbeitet ohne Rückführung.
- ti = 0: Der Regler arbeitet ohne Rückführung.
- t_proc = 0: Der Regler arbeitet mit einem voreingestellten Wert von t_proc = 60 s.

SERVO: Steuerung für elektrische Servomotoren

51

Übersicht

Einleitung

Dieses Kapitel beschreibt den Baustein SERVO.

Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Kurzbeschreibung	466
Darstellung	466
Parametrierung	468
Algorithmen zum Funktionsbaustein SERVO	470
Betriebsarten	471
Beispiele zum Funktionsbaustein SERVO	472
Laufzeitfehler	479

Kurzbeschreibung

Funktionsbeschreibung

Dieser Funktionsbaustein ermöglicht es, eine PID-Regelung an elektrischen Servomotoren mit oder ohne Stellungsrückführung vorzunehmen. Er wird dem Regler (PIDFF, PI_B) nachgeschaltet, dessen Digitalausgang er in die beiden logischen Ausgänge RAISE und LOWER umwandelt.

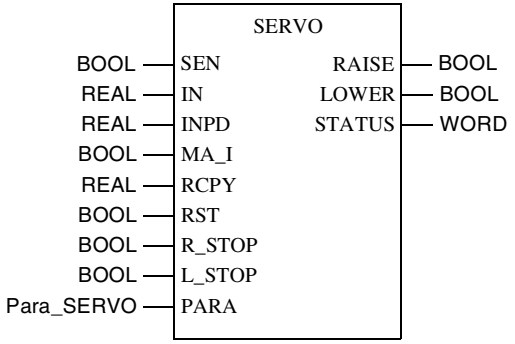
Wenn der Funktionsbaustein die Stellungsrückführung verwendet, wird eine Lageregelung des Stellgliedes realisiert. Wenn die Stellungsrückführung nicht verwendet wird, führen Regler und SERVO-Funktionsbaustein gemeinsam eine kontinuierliche (astatische) Regelung durch.

Als zusätzliche Parameter können EN und ENO projektiert werden.

Darstellung

Symbol

Darstellung des Bausteins



Parameterbeschreibung SERVO

Beschreibung der Bausteinparameter

Parameter	Datentyp	Bedeutung
SEN	BOOL	"1" : Einbeziehung eines neuen Wertes an den Eingängen INPD oder IN "0" : keine Einbeziehung der neuen Werte von INPD oder IN
IN	REAL	Reglerausgang OUT (0 bis 100%)
INPD	REAL	Ausgangsänderung OUTD des Reglers (-100% bis 100%)
MA_I	BOOL	Regler-Betriebsart (Ausgang MA_O) "1" : Automatikbetrieb "0" : Hand- oder Tracking-Betrieb
RCPY	REAL	Stellungsrückführung (0 bis 100%)
RST	BOOL	"1" : Neuinitialisierung des Funktionsbausteins (Nullsetzung der Ausgänge und des internen Bausteinstatus)
R_STOP	BOOL	Endstellung RAISE erreicht
L_STOP	BOOL	Endstellung LOWER erreicht
PARA	Para_SERVO	Parameter
RAISE	BOOL	Logischer Ausgang in Laufrichtung RAISE
LOWER	BOOL	Logischer Ausgang in Laufrichtung LOWER
STATUS	WORD	Statuswort

Parameterbeschreibung Para_SERVO

Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
en_rcpy	BOOL	"1" : Funktion mit Stellungsrückführung (Einbeziehung von RCPY)
rcpy_rev	BOOL	"1" : Invertierung von RCPY "0" : keine Invertierung von RCPY
t_motor	TIME	Öffnungszeit des Stellgliedes
t_mini	TIME	Mindestimpulsdauer

Parametrierung

Übersicht zur Parametrierung

Nachfolgend werden die folgenden Arbeitsweisen des Funktionsbausteins erklärt:

- *Mit Stellungsrückführung (en_rcpy = 1), S. 468*
- *Ohne Stellungsrückführung (en_rcpy = 0), S. 468*
- *Öffnungszeit des Stellgliedes (t_motor), S. 468*
- *Mindestimpulsdauer (t_mini), S. 468*
- *Abtastung / Parameter SEN, S. 469*
- *Erfassung der Endstellung, S. 469*

Mit Stellungsrückführung (en_rcpy = 1)

Wenn die Stellungsrückführung RCPY (en_rcpy = 1) verwendet wird, muß der Eingang IN an den Absolutwertausgang OUT eines Reglers angeschlossen werden (Regelbereich 0 bis 100%). Für jeden vom Regler generierten neuen Wert des Ausgangs OUT generiert der SERVO-Funktionsbaustein einen Diskretausgang RAISE oder LOWER, dessen Dauer proportional zur Abweichung IN - RCPY ist. Um eine richtige Funktionsweise des Funktionsbausteins zu gewährleisten, muß der Eingang MA_I an den Ausgang MA_O des Reglers angeschlossen werden.

Der Wert des RCPY-Eingangs kann einem Öffnungsprozentsatz (mit rcpy_rev = 0) oder einem Schließungsprozentsatz entsprechen (rcpy_rev auf 1 setzen).

Ohne Stellungsrückführung (en_rcpy = 0)

Wenn keine Stellungsrückführung angelegt wird (en_rcpy = 0), ist der INPD-Eingang an die Ausgangsänderung OUTD eines Reglers anzuschließen (Regelbereich -100 bis 100%). Für jeden neuen, vom Regler generierten OUTD-Wert generiert der Funktionsbaustein SERVO einen diskreten Ausgang RAISE oder LOWER, dessen Dauer proportional zur Ausgangsänderung des Reglers INPD ist. In diesem Fall muß der Eingang MA_I unbedingt an den Ausgang MA_O desselben Reglers angeschlossen werden, da der Algorithmus je nach Betriebsart leicht abweicht (siehe Abschnitt "*Algorithmen zum Funktionsbaustein SERVO, S. 470*").

Öffnungszeit des Stellgliedes (t_motor)

Der Parameter t_motor ermöglicht es dem Funktionsbaustein, sich auf die verschiedenen Servomotoren einzustellen.

Die RAISE oder LOWER aufzuschaltende Impulsdauer muß zur Öffnungszeit des Stellgliedes bei vollem Regelbereich proportional sein.

Mindestimpulsdauer (t_mini)

Über den Parameter t_mini kann vermieden werden, daß zu kurze Impulse generiert werden, die den Stellgliedern schaden können. Wenn die berechnete, RAISE oder LOWER aufzuschaltende Impulsdauer unterhalb von t_mini liegt, generiert der Funktionsbaustein keinen Impuls. Jeder bereits begonnene Impuls dauert mindestens t_mini.

Abtastung / Parameter SEN

In der Betriebsart Automatik wird die Auflösung der über den Funktionsbaustein SERVO durchgeführten Regelung ausgedrückt durch das Verhältnis (Abtastperiode des Regelkreises /Ausführungsperiode des Funktionsbausteins SERVO).

Der Regler muß also vor dem SERVO-Funktionsbaustein abgetastet werden (über einen SAMPLETM-Funktionsbaustein), der Funktionsbaustein SERVO muß aber bei jedem Zyklus ausgeführt werden. Im gegenteiligen Fall (wenn der Reglerbaustein taktgleich zum SERVO-Baustein ausgeführt wird), wird eine ungenaue Zweipunktregelung durchgeführt, die das Stellglied stark beansprucht.

Der Eingang SEN des Funktionsbausteins SERVO zeigt diesem an, ob der PID-Regelbaustein im laufenden Zyklus ausgeführt wurde oder nicht.

Der SEN-Eingang ermöglicht es also festzustellen, ob der Regler einen neuen Ausgang generiert hat oder nicht, so daß nicht mehrmals derselbe Ausgang berücksichtigt wird:

SEN =	**Bedeutung**
1	Einbeziehung eines neuen Wertes
0	keine Einbeziehung eines neuen Wertes

Wenn der Regler über den Funktionsbaustein SAMPLETM abgetastet wird, wie dies üblicherweise der Fall ist, reicht es aus, den SEN-Eingang des SERVO-Bausteins an den SAMPLETM-Ausgang anzuschließen (siehe Abschnitt "*Beispiele zum Funktionsbaustein SERVO, S. 472*").

Erfassung der Endstellung

Wenn eine Endstellung erfaßt wurde (R_STOP = 1 oder L_STOP = 1), wird der entsprechende Ausgang (RAISE bzw. LOWER) auf 0 geforct.

Algorithmen zum Funktionsbaustein SERVO

Algorithmus ohne Stellungsrückführung

In diesem Fall ermöglicht der dem Regler aufgeschaltete SERVO-Funktionsbaustein eine astatische Regelung. Der Algorithmus verwendet nicht den Absolutwertausgang OUT des Reglers, sondern die Ausgangsänderung OUTD. Der Ausgang RAISE (oder LOWER, je nach Vorzeichen der Änderung) wird für eine bestimmte Zeit auf 1 gesetzt. Diese Zeit ist proportional zur Öffnungszeit des Ventils (t_motor) und zum Änderungswert OUTD.

Die Formel gibt für die Dauer der dem Ausgang aufzuschlagenden Impulse (T_IMP) einen ersten theoretischen Wert vor:

$$T_IMP = OUTD(\%).t_motor$$

Weiterhin gilt für T_IMP (die Dauer der dem Ausgang aufzuschlagenden Impulse) Folgendes:

Wenn...	**Dann...**
T_IMP < t_mini	generiert der Baustein keinen Impuls, sondern speichert den Wert für die folgende Berechnung. Das ermöglicht die korrekte Verarbeitung von Regelapplikationen, bei denen die Ausgangsänderungen des Reglers schwach, aber dauerhaft sind. Um keine zu kurzen Impulse zu generieren, sind die dem Ausgang aufzuschlagenden Impulse auf eine Mindestdauer t_mini begrenzt.
der PID-Regler im Handbetrieb ist,	wird T_IMP kontinuierlich, bei jedem Zyklus berechnet. Die Berechnung berücksichtigt die vorher berechneten, aber noch nicht angelegten Zeitdauern mit einer Begrenzung auf t_motor. So kann eine beliebige Ausgangsänderung des PID-Reglers berücksichtigt werden, auch wenn der Impuls mehrere Zyklen dauert.
der PID-Regler im Automatikbetrieb ist,	berechnet der Funktionsbaustein SERVO den Parameter T_IMP immer dann neu, wenn der Regler seinen Ausgang aktualisiert, d.h. immer dann, wenn SEN auf 1 gesetzt wird. In dieser Betriebsart werden die vorher berechneten Zeitdauern nicht mehr berücksichtigt.

Algorithmus mit Stellungsrückführung

Der Algorithmus ähnelt stark dem vorhergehenden Fall.

Anstelle der Ausgangsänderung des PID-Reglers verwendet der SERVO-Funktionsbaustein jedoch die Abweichung zwischen dem Absolutwertausgang des PID-Reglers und der Stellungsrückführung (IN - RCPY).

Der Funktionsbaustein führt eine Lageregelung durch, wobei der Ausgang des PID-Reglers dem Sollwert entspricht und die Stellungsrückführung RCPY dem Istwert.

Im Gegensatz zum Algorithmus ohne Stellungsrückführung speichert der Funktionsbaustein im Handbetrieb die vorher berechneten, aber den Ausgängen RAISE und LOWER noch nicht aufgeschalteten Zeitdauern nicht.

Betriebsarten

Anpassung an Betriebsarten

Der Eingang MA_I ermöglicht es dem SERVO-Funktionsbaustein, sein Verhalten der Betriebsart des Reglers anzupassen. Hierzu muß er an den Ausgang MA_O des Reglers oder des entsprechenden MS-Funktionsbausteins angeschlossen werden.

Betriebsart Automatik

Der Funktionsbaustein SERVO liest den Reglerausgang nur dann erneut, wenn dieser aktualisiert wurde (d.h. immer dann, wenn SEN auf 1 gesetzt wird).

Betriebsart Hand

Hier kann der Reglerausgang vom Anwender jederzeit geändert werden. Damit ein neuer Wert so schnell wie möglich einbezogen wird, liest der Funktionsbaustein den Reglerausgang bei jedem Zyklus.

In dieser Betriebsart kann der Anwender per Hand die an den OUT-Ausgang eines Reglers oder eines MS-Bausteins angeschlossene Variable ändern. Wenn keine Stellungsrückführung verwendet wird, kann diese Variable die Endstellung (100% oder 0%) annehmen, auch wenn das Stellglied keine seiner beiden Endstellungen angenommen hat. Es ist jedoch noch möglich, die Ausgangsänderung OUTD per Hand zu ändern, indem der Ausgang OUT des Funktionsbausteins MS auf über 100% eingestellt wird (oder auf weniger als 0%). Der für OUT eingegebene Wert wird hierbei zur Berechnung von OUTD verwendet, bevor er erneut begrenzt wird.

Beispiele zum Funktionsbaustein SERVO

Beispiel-Übersicht

In diesem Abschnitt wird der Einsatz des Funktionsbausteins SERVO in folgenden Beispielen gezeigt:

- *Beispiel Betriebsart Automatik mit Stellungsrückführung, S. 472*
- *Beispiel Betriebsart Automatik ohne Stellungsrückführung im Handbetrieb, S. 476*

Beispiel Betriebsart Automatik mit Stellungsrück-führung

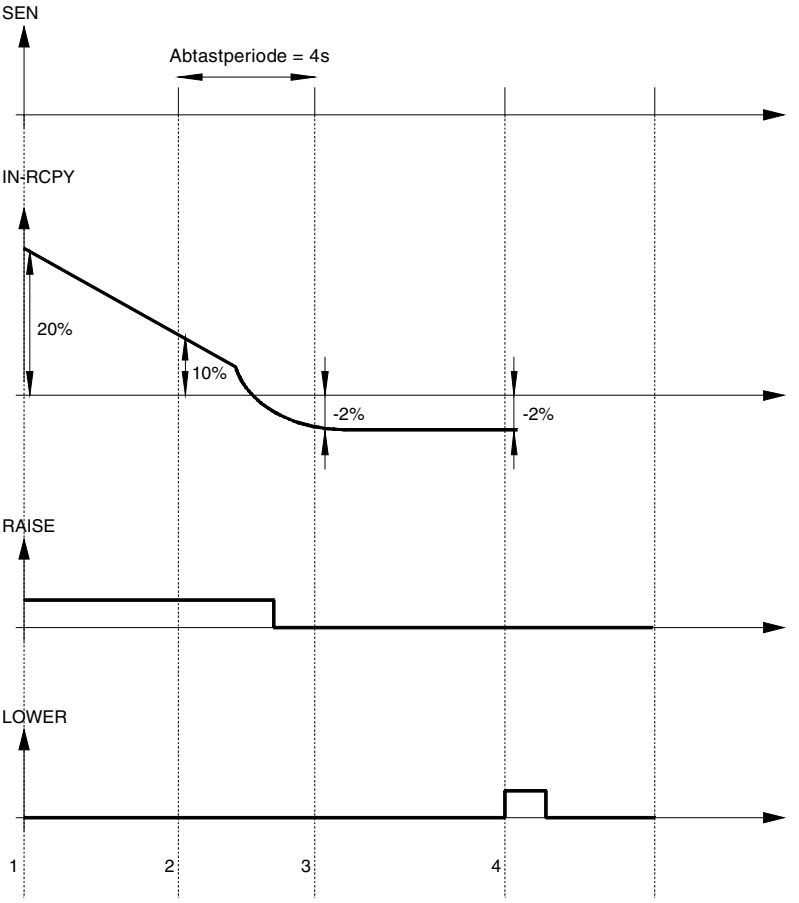
Im Beispiel wird das Verhalten des Funktionsbausteins in der Betriebsart Automatik mit Stellungsrückführung dargestellt. Immer, wenn der SEN-Eingang auf 1 gesetzt wird (im Beispiel alle 4 s), berücksichtigt der Funktionsbaustein SERVO einen neuen Abweichungswert IN-RCPY.

Es gelten folgende Parametervorgaben:

Parameter	Vorgabe
t_motor	25 s
t_mini	1 s
Abtastperiode	4 s

Taktdiagramm (Automatik mit Stellungsrück-führung)

Taktdiagramm zur Betriebsart Automatik mit Stellungsrückführung



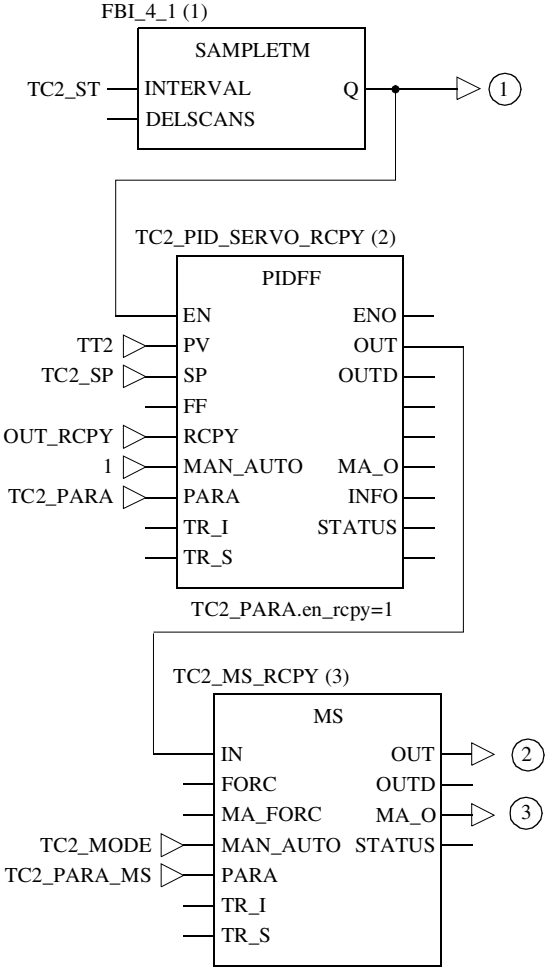
Erläuterung der Takte

Erklärung der markierten Stellen:

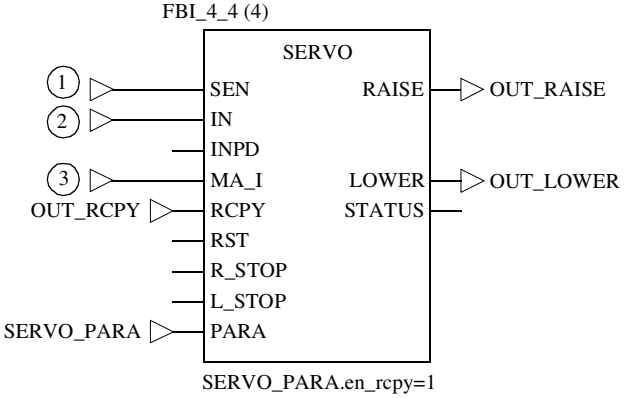
Stelle Nr.	Erklärung
1	Die Abweichung IN-RCPY beträgt 20%: ein Impuls einer Dauer von 5 s (=20% von 25 s) wurde am RAISE-Ausgang generiert.
2	Die Abweichung beträgt nur noch 10%, ein Impuls von 2,5 s (= 10% von 25 s) wurde am RAISE-Ausgang generiert, die Sekunde, die vom vorherigen Impuls übrig bleibt, wird nicht berücksichtigt.
3	Die Abweichung beträgt jetzt -2%, was einem Impuls von 0,5 s an LOWER. entspricht.Da t_mini 1 s entspricht, wird kein Impuls generiert (die Zeitdauer von 0,5 s wird jedoch gespeichert).
4	Die Abweichung beträgt immer noch -2%, aber der entsprechende Impuls (0,5 s) wird zum vorher gespeicherten Impuls addiert, um 1 s zu erreichen. Die Dauer entspricht t_mini, der Impuls wird daher dem LOWER-Ausgang aufgeschaltet.

Programmier-beispiel (Automatik mit Stellungsrück-führung)

Darstellung des Funktionsplans, Teil 1



Darstellung des Funktionsplans, Teil 2



OUT_RCPY Istwert der Ventilstellungsrückführung

Beispiel Betriebsart Automatik ohne Stellungsrück-führung im Handbetrieb

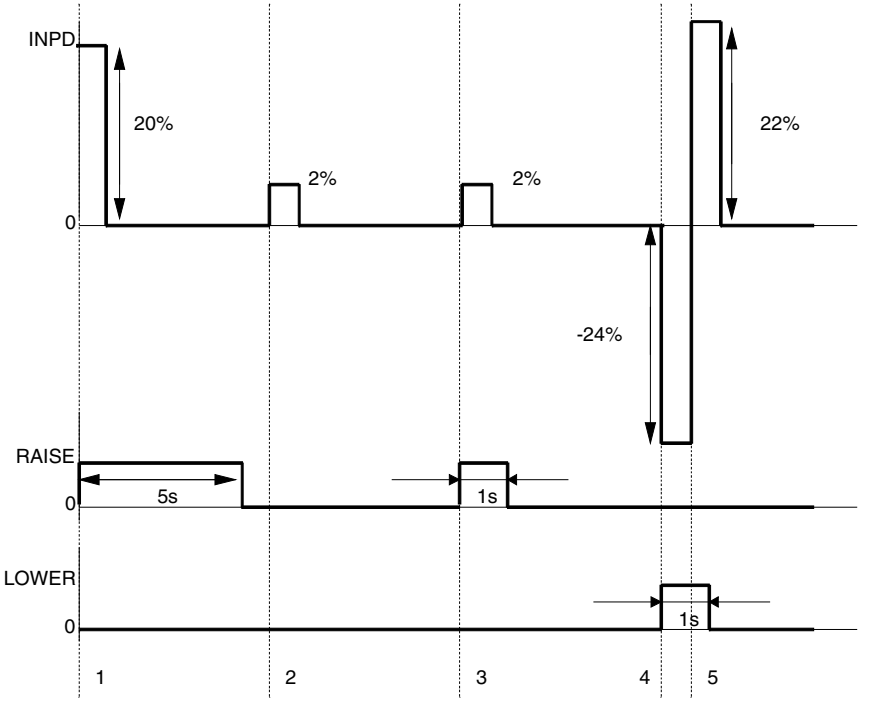
Im Besipiel wird das Verhalten des Funktionsbausteins in der Betriebsart Automatik ohne Stellungsrückführung im Handbetrieb dargestellt. In diesem Fall wird der Wert von INPD bei jeder Ausführung des Funktionsbausteins SERVO berücksichtigt, unabhängig davon, welchen Wert der Eingang SEN besitzt

Es gelten folgende Parametervorgaben:

Parameter	Vorgabe
t_motor	25 s
t_mini	1 s

Taktdiagramm (Automatik ohne Stellungsrück-führung)

Betriebsart Automatik ohne Stellungsrückführung im Handbetrieb



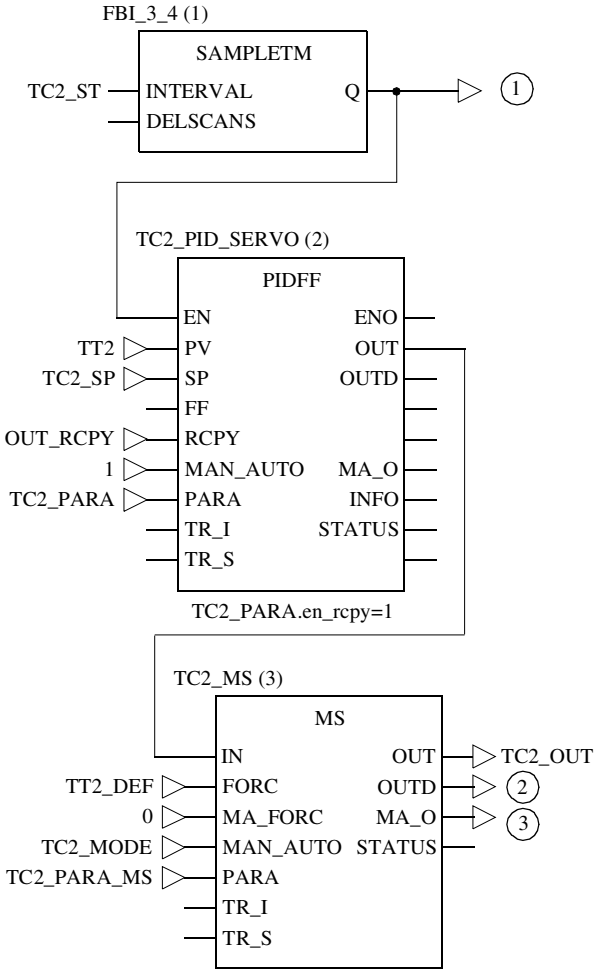
Erläuterung der Takte

Erklärung der markierten Stellen:

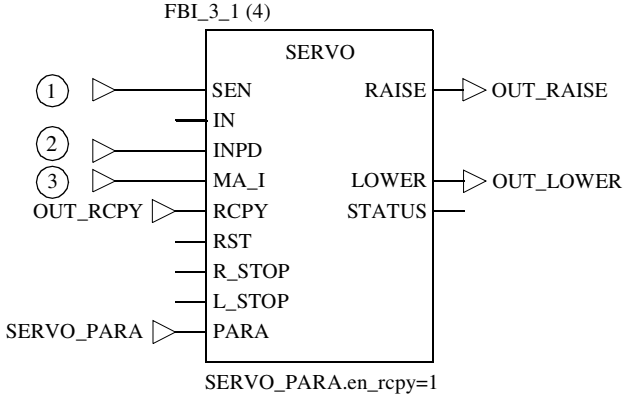
Stelle Nr.	Erklärung
1	Die Änderung des PID-Reglerausgangs beträgt + 20%, in diesem Fall betrifft der Impuls den RAISE-Ausgang und dauert 5 s (= 20% von 25 s).
2	Die Änderung des PID-Reglers beträgt +2 %, was einer Impulsdauer von 0,5 s entspricht. Der Impuls ist kleiner als t_mini (=1 s.), beeinflußt also nicht die Ausgänge.
3	Bei der zweiten Änderung von +2 % addiert die Funktion diese Änderung zur vorhergehenden (die einer Abweichung entspricht, die unterhalb des Mindestwertes liegt), was einer positiven Gesamtänderung von +4 % entspricht, d.h. einem Impuls von 1 s am RAISE-Ausgang.
4	Bei einer Änderung von -24 % beträgt der Impuls am LOWER-Ausgang 6 s
5	Vor Ablauf der Folgesekunde führt eine weitere Änderung von + 22 % zu einer Gesamtsystemänderung von 2 % < Änderung von t_mini (4 %). Die Funktion beendet den Mindestimpuls von 1 s.

Programmier-beispiel (Automatik ohne Stellungsrück-führung)

Darstellung des Funktionsplanes, Teil 1



Darstellung des Funktionsplanes, Teil 2



TT2_DEFF Fehlermerker des Istwertes TT2: Wenn TT2 fehlerhaft ist, wird der Regelkreis auf Handbetrieb geforct.

Laufzeitfehler

Statuswort

Im Statuswort werden folgende Meldungen angezeigt:

Bit	Bedeutung
Bit 0 = 1	Fehler in einer Berechnung mit Gleitkommawerten
Bit 1= 1	Erfassung eines unzulässigen Wertes an einem der Gleitkommawerteingänge
Bit 2= 1	Division durch Null bei einer Berechnung mit Gleitkommawerten
Bit 3 = 1	Kapazitätsüberlauf bei Berechnung in Gleitkommawerten
Bit 4 = 1	IN bzw. RCPY liegen nicht im Bereich [0, 100] oder INPD liegt außerhalb des Bereichs [-100, 100]. Zur Berechnung verwendet der Funktionsbaustein einen Wert, der auf den nächstliegenden korrekten Wert begrenzt ist, d.h. je nach Fall auf den Wert 0, 100 oder -100.

Fehlermeldung

Ein Fehler erscheint, wenn an einem Eingang ein nicht gleitender Wert erfaßt wird oder wenn bei einer Berechnung mit Gleitkommawerten ein Problem auftritt. Die Ausgänge RAISE und LOWER werden in diesem Fall auf Null gesetzt.-

SMOOTH_RATE: Differenzierer mit Glättung

52

Übersicht

Einleitung

Dieses Kapitel beschreibt den Baustein SMOOTH_RATE.

Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Kurzbeschreibung	482
Darstellung	482
Formeln zum Funktionsbaustein SMOOTH_RATE	483
Detailbeschreibung	484

Kurzbeschreibung

Funktionsbeschreibung

Dieser Funktionsbaustein stellt ein Differenzierglied mit einem mit der Zeitkonstante LAG verzögerten Ausgang Y dar.

Der Funktionsbaustein besitzt folgende Betriebsarten:

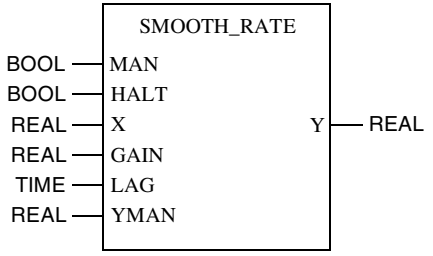
- Hand
- Halt
- Automatik

Als zusätzliche Parameter können EN und ENO projektiert werden.

Darstellung

Symbol

Darstellung des Bausteins



Parameterbeschreibung

Beschreibung der Bausteinparameter

Parameter	Datentyp	Bedeutung
MAN	BOOL	"1" = Betriebsart Hand
HALT	BOOL	"1" = Betriebsart Halt
X	REAL	Eingangsgröße
GAIN	REAL	Verstärkung der Differenzierung
LAG	TIME	Verzögerungszeitkonstante
YMAN	REAL	Hand-Stellwert
Y	REAL	Ausgang Differenzierer mit Glättung

Formeln zum Funktionsbaustein SMOOTH_RATE

Übertragungs-funktion

Die Übertragungsfunktion für Y lautet:

$$G(s) = GAIN \times \frac{1}{1 + s \times LAG}$$

Ausgang Y

Der Ausgang Y wird wie folgt ermittelt:

$$Y = \frac{dt}{dt + LAG} \times (Y_{(old)} + GAIN \times (X_{(new)} - X_{(old)}))$$

Erläuterung der Formelgrößen

Bedeutung der Größen in den obigen Formeln:

Größe	Bedeutung
dt	Zeitdifferenz zwischen aktuellem Zyklus und vorherigem Zyklus
$X_{(new)}$	Wert des Eingangs X aus dem aktuellen Zyklus
$X_{(old)}$	Wert des Eingangs X aus dem vorherigen Zyklus
$Y_{(old)}$	Wert des Ausgangs Y aus dem vorherigen Zyklus

Detailbeschreibung

Parametrierung

Die Parametrierung des Funktionsbausteins erfolgt durch Festlegen der Verstärkung GAIN des Differenzierers und der Zeitkonstante LAG, mit welcher der Ausgang Y verzögert wird.

Für sehr kleine Abtastzeiten und dem Einheitssprung am Eingang X (Sprung am Eingang X von 0 auf 1.0) wird der Ausgang Y auf den Wert GAIN springen (theoretischer Wert - real ein bißchen kleiner, bedingt durch die nicht unendlich kleine Abtastzeit), um dann mit der Zeitkonstante LAG wieder auf 0 zurückzugehen.

Betriebsarten

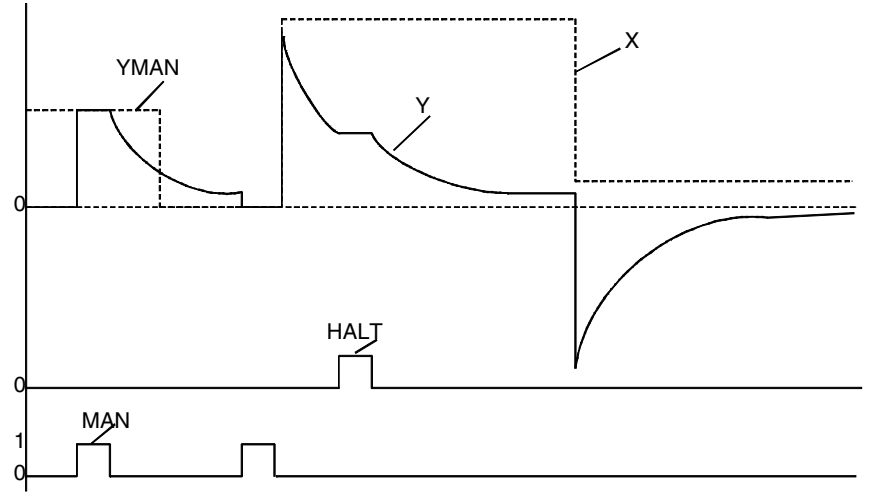
Der Funktionsbaustein SMOOTH_RATE hat 3 Betriebsarten: Automatik, Hand und Halt.

Diese Betriebsarten können über die Eingänge MAN und HALT ausgewählt werden:

Betriebsart	MAN	HALT	Bedeutung
Automatik	0	0	Der Funktionsbaustein wird, wie in "Parametrierung" beschrieben, bearbeitet.
Hand	1	0 oder 1	Der Eingang YMAN wird auf den Ausgang Y übertragen.
Halt	0	1	Der Ausgang Y wird auf dem zuletzt berechneten Wert gehalten.

Beispiel

Im nachfolgenden Bild ist die Sprungantwort des Funktionsbausteins SMOOTH_RATE mit GAIN = 1 und LAG = 10 s dargestellt:



SP_SEL: Sollwertschalter

53

Übersicht

Einleitung

Dieses Kapitel beschreibt den Baustein SP_SEL.

Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Kurzbeschreibung	488
Darstellung	489
Detailbeschreibung	491
Laufzeitfehler	494

Kurzbeschreibung

Funktionsbeschreibung

Dieser Funktionsbaustein ermöglicht die Auswahl des Sollwerttyps, der im Regelkreis verwendet wird:

Sollwerttyp	Erläuterung
Externer Sollwert (SP_RSP = 1)	Der Sollwert stammt aus einer bausteinexternen Berechnung über den Eingang RSP (Remote Set Point). Der Eingangswert RSP wird auf den SP-Ausgang geführt.
Interner Sollwert (SP_RSP = 0)	Der Sollwert muß direkt vom Anwender geändert werden (Local Set Point). In dieser Betriebsart wird der Ausgang SP nicht über den Funktionsbaustein eingestellt, der Bediener ändert die an SP angeschlossene Variable selbst.

Als zusätzliche Parameter können EN und ENO projektiert werden.

Eigenschaften

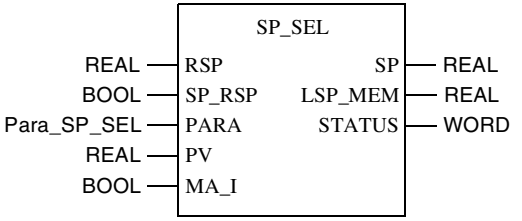
Der Funktionsbaustein SP_SEL besitzt folgende Eigenschaften:

- Die Umschaltung zwischen den beiden Sollwerttypen kann stoßfrei erfolgen
- Betrieb mit Nachführung des Sollwertes, wenn der Regler im Handbetrieb ist
- Obere und untere Begrenzung des verwendeten Sollwerts

Darstellung

Symbol

Darstellung des Bausteins



Parameterbeschreibung SP_SEL

Beschreibung der Bausteinparameter

Parameter	Datentyp	Bedeutung
RSP	REAL	Externer Sollwert (Remote Set Point)
SP_RSP	BOOL	Vom Regler verwendeter Sollwerttyp: "1" : Externer Sollwert "0" : Interner Sollwert
PARA	Para_SP_SEL	Parameter
PV	REAL	Zu regelnde Größe
MA_I	BOOL	Betriebsart des verbundenen Reglers "1" : Betriebsart Automatik "0" : Betriebsart Hand
SP	REAL	Vom Regler verwendeter Sollwert
LSP_MEM	REAL	Speicher für internen Sollwert (Local Set Point MEMory)
STATUS	WORD	Statuswort

Parameterbeschreibung Para_SP_SEL

Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
sp_min	REAL	Unterer Grenzwert für verwendeten Sollwert
sp_max	REAL	Oberer Grenzwert für verwendeten Sollwert
bump	BOOL	Bei Umschaltung extern/ intern: "1" : der SP-Ausgang wird mit dem Wert von LSP_MEM geforct "0" : stoßfreie Umschaltung
track	BOOL	"1" : im Handbetrieb wird der Wert von SP an PV angeglichen (nur interner Sollwert)
rate	REAL	Steigung von SP bei Umschaltung intern /extern in Einheiten pro Sekunde (≥0)

Detailbeschreibung

Umschaltung des Sollwertes

Die Umschaltung des Sollwertes kann in zwei Richtungen erfolgen:

Wenn...	Dann ...
SP_RSP von 0 → 1	erfolgt die Umschaltung des internen Sollwertes auf einen externen Sollwert
SP_RSP von 1 → 0	erfolgt die Umschaltung des externen Sollwertes auf den internen Sollwert

SP_RSP von 0 → 1

Die Umschaltung des internen Sollwertes auf einen externen Sollwert erfolgt stoßfrei: der Wert des SP-Ausgangs paßt sich zunehmend dem externen Sollwert RSP an, wobei er die Rampe rate beschreibt.

Wenn rate = 0, wird keine Rampe gefahren und SP ist die direkte Kopie von RSP.

SP_RSP von 1 → 0

Die Umschaltung des externen Sollwertes auf den internen Sollwert erfolgt in Abhängigkeit vom Element bump auf zwei Arten:

Wenn...	Dann ...
bump = 0	erfolgt die Umschaltung stoßfrei: Der Funktionsbaustein hört auf, den RSP-Eingang auf den SP-Ausgang zu kopieren. Der interne Sollwert SP entspricht dann dem letzten externen Sollwert RSP vor Umschaltung. Der Anwender kann diesen dann ändern. In diesem Fall ist es nicht nötig, den Ausgang LSP_MEM anzuschließen.
bump = 1	wird der Wert des Ausgangs LSP_MEM bei der Umschaltung auf den SP-Ausgang geführt (hierbei kann es zu Stößen kommen). Die Wertvorgabe für LSP_MEM entspricht dem letzten Sollwert SP, bevor der Funktionsbaustein in externen Betrieb übergeht. Um den internen Betrieb mit einem anderen Sollwert neu zu starten, reicht es aus, LSP_MEM zu ändern, solange der Baustein noch im externen Betrieb ist (weitere Details siehe "*Aufgabe des Ausgangs LSP_MEM, S. 492*").

Nachgeführter Sollwert (track = 1)

Wenn beim internen Sollwert (SP_RSP=0) der verbundene Regler im Handbetrieb ist, ermöglicht diese Betriebsart eine stetige Kopie des PV-Eingangs auf den verwendeten Sollwert SP. Das ermöglicht eine stoßfreie Umschaltung Hand/ Automatik (Es ist ebenfalls möglich, daß der Regler das stoßfreie Verhalten selbst regelt).

Bei dieser Betriebsart müssen die Eingänge PV und MA_I des Funktionsbausteins SP_SEL angeschlossen sein. Sie müssen dieselben Werte annehmen wie der PV-Eingang des Reglers und dessen Ausgang MA_O. Wenn track = 0, brauchen diese Eingänge nicht angeschlossen zu werden.

Begrenzungen

Bei jeder Betriebsart (extern oder intern) wird der verwendete Sollwert SP auf den Bereich zwischen sp_min und sp_max begrenzt.

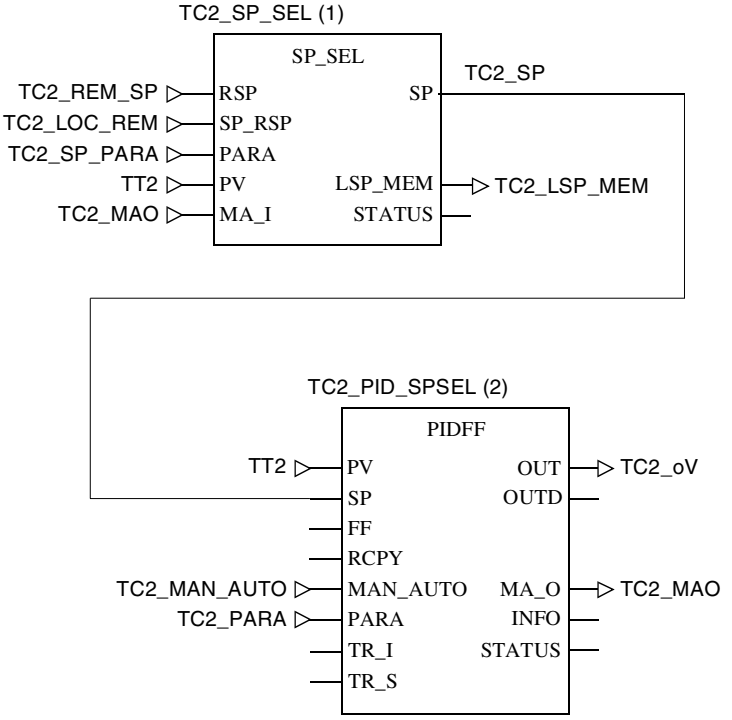
Aufgabe des Ausgangs LSP_MEM

Dieser Ausgang ermöglicht es dem Anwender, den Sollwert SP bei der Umschaltung extern - intern zu steuern:

Art des Sollwertes	Verhalten des Ausgangs
Interner Sollwert	Der Wert von SP wird stetig auf LSP_MEM geführt.
Umschaltung auf externen Sollwerte	Der Wert von LSP_MEM wird vom Baustein nicht mehr geändert und behält daher den letzten verwendeten internen Sollwert bei.
Rückkehr zum internen Sollwert	Hierzu gibt es drei Möglichkeiten: **1.** bump = 0: Der letzte externe Sollwert wird zugrunde gelegt, in diesem Fall braucht LSP_MEM nicht angeschlossen zu werden). **2.** bump = 1: Der letzte gespeicherte interne Sollwert wird zugrunde gelegt, bei der Umschaltung kopiert der Baustein den Wert von LSP_MEM auf SP. **3.** Der Funktionsbaustein kann den internen Betrieb über einen beliebigen, vom Anwender gewählten Wert starten. Wenn der Wert der an LSP_MEM angeschlossenen Variablen vor Übergang auf den internen Sollwert (mit bump = 1) geändert wird, wird bei der Umschaltung dieser Wert auf SP geführt.

Beispiel zur Programmierung

Nachfolgend finden Sie ein Beispiel zur Programmierung des Funktionsbausteins SP_SEL



TC2_SP wird vom Operator in der Betriebsart "interner Sollwert" eingegeben.

Laufzeitfehler

Statuswort

Im Statuswort werden folgende Meldungen angezeigt:

Bit	Bedeutung
Bit 0 = 1	Fehler in einer Berechnung mit Gleitkommawerten
Bit 1= 1	Erfassung eines unzulässigen Wertes an einem der Gleitkommaeingänge
Bit 2= 1	Division durch Null bei einer Berechnung in Gleitkommawerten
Bit 3 = 1	Kapazitätsüberlauf bei einer Berechnung in Gleitkommawerten
Bit 4 = 1	rate ist negativ : zur Berechnung verwendet der Funktionsbaustein den Wert 0
Bit 5 = 1	Der Ausgang SP hat den unteren Grenzwert sp_min erreicht: SP wird auf sp_min geforct
Bit 6 = 1	Der Ausgang SP hat den oberen Grenzwert sp_max erreicht: SP wird auf sp_max geforct

Fehlermeldung

Ein Laufzeitfehler erscheint, wenn an einem Eingang ein nicht gleitender Wert erfaßt wird oder wenn bei einer Berechnung mit Gleitkommawerten ein Problem auftritt. Die Ausgänge SP und LSP_MEM bleiben unverändert.

Warnung

Eine Warnung wird ausgegeben, wenn rate negativ ist; der Baustein verwendet dann zur Berechnung den Wert 0.

SPLRG: Steuerung von 2 Stellgliedern

54

Übersicht

Einleitung

Dieses Kapitel beschreibt den Baustein SPLRG.

Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Kurzbeschreibung	496
Darstellung	497
Detailbeschreibung	498
Laufzeitfehler	500

Kurzbeschreibung

Funktionsbeschreibung

Der Einsatz dieses Funktionsbausteins ist dann sinnvoll, wenn 2 Stellglieder eingesetzt werden, um den gesamten Regelbereich abdecken zu können (bei 2 voneinander entfernten Arbeitspunkten: einer unten und einer oben).

Der Regler ist auch für Dreipunkt-Schrittregelungen geeignet, d.h. für solche Fälle, wo die beiden Stellglieder gegensätzlich wirken (ein Stellglied heizt, das andere kühlt).

Als zusätzliche Parameter können EN und ENO projektiert werden.

Eigenschaften

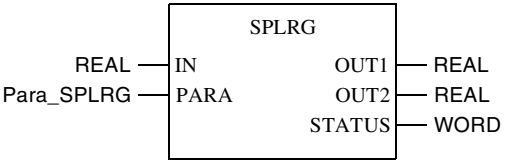
Der Funktionsbaustein SPLRG besitzt folgende Eigenschaften:

- Möglichkeit der Steuerung einer Unempfindlichkeitszone oder Übergangszone mit Angleichung der Eigenschaften der beiden Stellglieder
- Der IN-Eingang wird als Prozentsatz ausgedrückt (0-100%), die Ausgänge OUT1 und OUT2 werden in physikalischen Einheiten ausgedrückt.

Darstellung

Symbol

Darstellung des Bausteins



Parameterbeschreibung SPLRG

Beschreibung der Bausteinparameter

Parameter	Datentyp	Bedeutung
IN	REAL	Aufzulösender Wert (0 bis 100%)
PARA	Para_SPLRG	Parameter
OUT1	REAL	Stellgröße für Stellglied 1
OUT2	REAL	Stellgröße für Stellglied 2
STATUS	WORD	Statuswort

Parameterbeschreibung Para_SPLRG

Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
out1_th1	REAL	Eingangswert IN, für den gilt: OUT1 = out1_inf
out1_th2	REAL	Eingangswert IN, für den gilt: OUT1 = out1_sup
out1_inf	REAL	Unterer Grenzwert des Ausgangs OUT1
out1_sup	REAL	Oberer Grenzwert des Ausgangs OUT1
out2_th1	REAL	Eingangswert IN, für den gilt: OUT2 = out2_inf
out2_th2	REAL	Eingangswert IN, für den gilt: OUT2 = out2_sup
out2_inf	REAL	Unterer Grenzwert für Ausgang OUT2
out2_sup	REAL	Oberer Grenzwert für Ausgang OUT2

Detailbeschreibung

Parametrierung

Die Parametrierung des Funktionsbausteins besteht in der Definition der Eigenschaften jedes Stellgliedes, d.h. in der Art der Gradientenänderung der beiden Reglerausgänge im Verhältnis zum Eingang IN.

Für den Ausgang OUT1 sind folgende Punkte zu definieren:

Element	Bedeutung
out1_inf	Unterer Bereichs-Grenzwert
out1_sup	Oberer Bereichs-Grenzwert
out1_th1	Schwellwert, d.h. der Eingangswert IN, für den gilt: Ausgang OUT1 = out1_inf
out1_th2	Schwellwert, d.h. der Eingangswert IN, für den gilt: Ausgang OUT1 = out1_sup

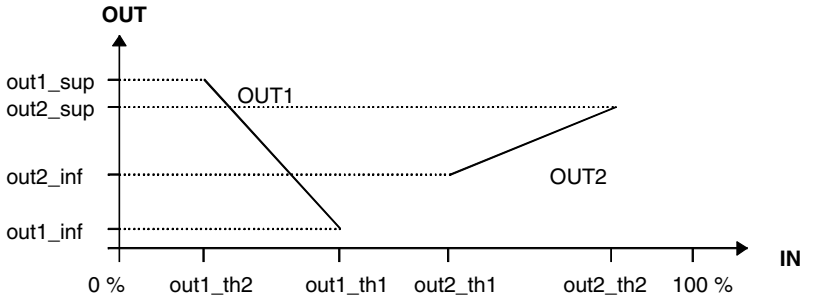
Bei den beiden Schwellwerten ist die Änderung des Wertes von OUT1 linear. Abgesehen von den beiden Schwellwerten kann keine Ausgangsänderung mehr auftreten, diese ist begrenzt auf out1_inf bzw. out1_sup.

Je nach Einstellung der beiden Schwellwerte, sind die Regeleigenschaften durch eine positive (bei out1_th1 < out1_th2) oder eine negative Steigung (bei out1_th2 < out1_th1) gekennzeichnet.

Die nachfolgenden Diagramme stellen die Eigenschaften der beiden Stellglieder bei Split-Range- und Dreipunkt-Schrittregelung dar.

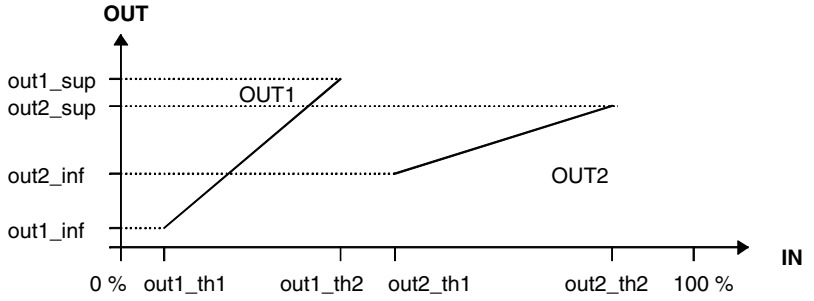
Dreipunkt-Schrittregelung

Hier finden Sie die Darstellung der Eigenschaften der beiden Stellglieder bei einer Dreipunkt-Schrittregelung



Split-Range-Regelung

Hier finden Sie die Darstellung der Eigenschaften der beiden Stellglieder bei einer Split-Range-Regelung



Hinweis: Die Ausgänge dieses Reglers können ohne Stellungsrückführung nicht zur Steuerung eines SERVO-Funktionsbausteins verwendet werden.

Betriebsarten

Der Funktionsbaustein SPLRG ist keiner spezifischen Betriebsart zugeordnet. Es ist jedoch durchaus möglich, die beiden Ausgänge des Funktionsbausteins per Hand zu steuern, indem jedem Ausgang ein MS-Funktionsbaustein aufgeschaltet wird. Bei der Programmierung sollte der Anwender darauf achten, daß die Rückkehr zur Betriebsart Automatik stoßfrei erfolgt.

Laufzeitfehler

Statuswort

Im Statuswort werden folgenden Meldungen angezeigt:

Bit	Bedeutung
Bit 0 = 1	Fehler in einer Berechnung mit Gleitkommawerten
Bit 1= 1	Erfassung eines unzulässigen Wertes an einem der Gleitkommawerteingänge
Bit 2= 1	Division durch Null bei einer Berechnung mit Gleitkommawerten
Bit 3 = 1	Kapazitätsüberlauf bei Berechnung in Gleitkommawerten
Bit 4 = 1	IN oder einer der Parameter out1_th1, out1_th2, out2_th1, out2_th2 liegt nicht im Bereich [0, 100]: zur Berechnung verwendet der Funktionsbaustein den Wert 0 bzw. 100.
Bit 5 = 1	Der Ausgang OUT1 hat den unteren Grenzwert out1_inf erreicht: OUT1 wird auf out1_inf geforct.
Bit 6 = 1	Der Ausgang OUT1 hat den oberen Grenzwert out1_sup erreicht: OUT1 wird auf out1_sup geforct.
Bit 7 = 1	Die beiden Schwellwerte eines Ausgangs sind identisch: out1_th1 = out1_th2, out2_th1 = out2_th2.
Bit 8 = 1	Der Ausgang OUT2 hat den unteren Grenzwert out2_inf erreicht: OUT2 wird auf out2_inf geforct.
Bit 9 = 1	Der Ausgang OUT2 hat den oberen Grenzwert out2_sup erreicht: OUT2 wird auf out2_sup geforct.

Fehlermeldung

Ein Laufzeitfehler wird in folgenden Fällen gemeldet:

- An einem Eingang liegt ein nicht gleitender Wert an
- Bei einer Berechnung mit Gleitkommawerten tritt ein Problem auf.
- Die beiden Schwellen desselben Ausgangs sind identisch: out1_th1 = out1_th2 bzw. out2_th1 = out2_th2

Die Ausgänge OUT1 und OUT2 bleiben immer unverändert.

Warnung

Eine Warnung wird ausgegeben, wenn einer der Parameter out1_th1, out1_th2, out2_th1, out2_th2 nicht im Bereich [0, 100] liegt. In diesem Fall verwendet der Funktionsbaustein zur Berechnung den Wert 0 bzw. 100.

STEP2: Zweipunktregler

55

Übersicht

Einleitung

Dieses Kapitel beschreibt den Baustein STEP2.

Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Kurzbeschreibung	502
Darstellung	503
Detailbeschreibung	504
Laufzeitfehler	506

Kurzbeschreibung

Funktionsbeschreibung

Dieser Funktionsbaustein ist geeignet für einfache Zweipunkt-Regelungen.

Die Steuerung des Stellgliedes erfolgt je nach Richtung der Abweichung Istwert-Sollwert in bezug auf den oberen und unteren Grenzwert.

Als zusätzliche Parameter können EN und ENO projektiert werden.

Eigenschaften

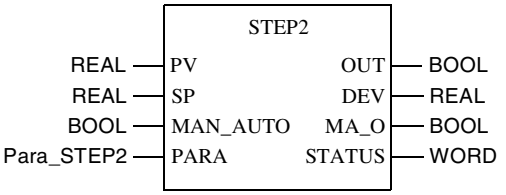
Der Reglerbaustein besitzt folgende Eigenschaften:

- Obere und untere Begrenzung des Sollwertes zwischen pv_inf und pv_sup
- Die Reglereingangswerte (Istwert, Sollwert und zugehörige Parameter) werden in physikalischen Einheiten ausgedrückt.

Darstellung

Symbol

Darstellung des Bausteins



Parameterbeschreibung STEP2

Beschreibung der Bausteinparameter

Parameter	Datentyp	Bedeutung
PV	REAL	Istwert (Process Value)
SP	REAL	Sollwert (Set Point)
MAN_AUTO	BOOL	Regler-Betriebsart: "1" : Automatik-Betrieb "0" : Halt-Betrieb
PARA	Para_STEP2	Parameter
OUT	BOOL	Logischer Ausgang
DEV	REAL	Abweichung (PV-SP)
MA_O	BOOL	Aktuelle Betriebsart des Funktionsbausteins (0: Halt, 1: Automatik)
STATUS	WORD	Statuswort

Parameterbeschreibung Para_STEP2

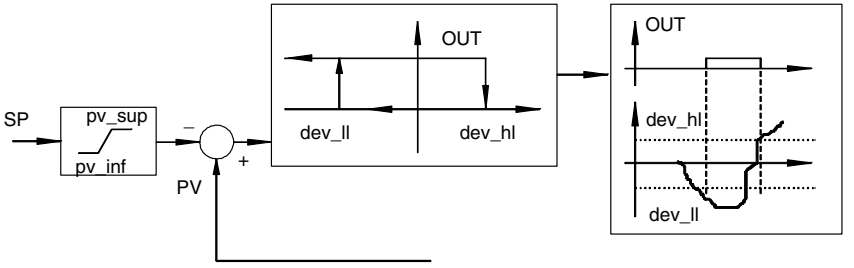
Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
dev_ll	REAL	Untere Schwelle der Abweichung (≤ 0)
dev_hl	REAL	Obere Schwelle der Abweichung (≥ 0)
pv_inf	REAL	Untere Grenze des Istwertbereichs
pv_sup	REAL	Obere Grenze des Istwertbereichs

Detailbeschreibung

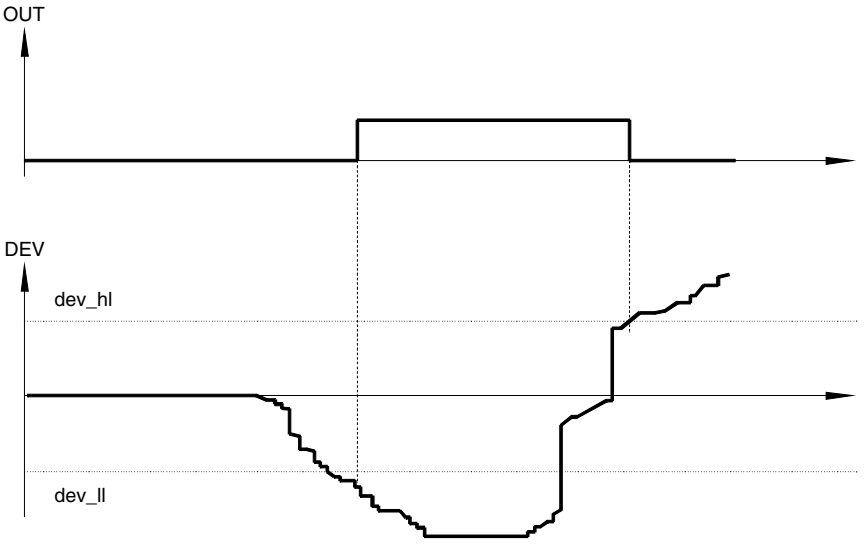
Strukturbild

Nachfolgend finden Sie das Strukturbild des Bausteins STEP2:



Verhalten des Ausgangs

Verhalten des Ausgangs OUT:



Wenn die Abweichung (DEV = PV - SP) die untere Schwelle dev_ll unterschreitet, wird der logische Ausgang OUT auf 1 gesetzt. Wenn die Abweichung jedoch erneut ansteigt, wird der Ausgang OUT nur dann nullgesetzt, wenn die Abweichung dev_hl übersteigt.

Hinweis: Um eine fehlerfreie Funktionsweise des Bausteins zu gewährleisten, darf der Ausgang OUT nicht invertiert werden.

Betriebsarten

Der Reglerbaustein STEP2 verfügt über 2 Betriebsarten, je nach Wert des Parameters MAN_AUTO :

Betriebsart	MAN_AUTO	Bedeutung
Automatik	1	Der Ausgang OUT wird vom Reglerbaustein selbst berechnet.
Halt	0	Der Ausgang OUT wird auf dem zuletzt berechneten Wert gehalten.

Laufzeitfehler

Statuswort

Im Statuswort werden folgende Meldungen angezeigt:

Bit	Bedeutung
Bit 0 = 1	Fehler in einer Berechnung mit Gleitkommawerten
Bit 1= 1	Erfassung eines unzulässigen Wertes an einem der Gleitkommawerteingänge
Bit 2= 1	Division durch Null bei einer Berechnung mit Gleitkommawerten
Bit 3 = 1	Kapazitätsüberlauf bei Berechnung in Gleitkommawerten
Bit 4 = 1	Folgende Verhalten werden angezeigt: ● SP liegt außerhalb des Bereichs [pv_inf, pv_sup]: SP wird begrenzt auf pv_inf bzw. pv_sup ● dev_ll > 0 bzw. dev_hl < 0: der Baustein verwendet den Wert 0

Fehlermeldung

Ein Laufzeitfehler erscheint, wenn an einem Eingang ein nicht gleitender Wert erfaßt wird oder wenn bei einer Berechnung mit Gleitkommawerten ein Problem auftritt. Der Ausgang OUT wird dann auf 0 gesetzt, die Ausgänge DEV und MA_O bleiben unverändert.

Warnung

Eine Warnung wird ausgegeben, wenn dev_ll > 0 bzw. dev_hl < 0 ist. In diesem Fall verwendet der Funktionsbaustein den Wert 0.

STEP3: Dreipunktregler

56

Übersicht

Einleitung

Dieses Kapitel beschreibt den Baustein STEP3.

Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Kurzbeschreibung	508
Darstellung	509
Detailbeschreibung	510
Laufzeitfehler	512

Kurzbeschreibung

Funktionsbeschreibung

Dieser Funktionsbaustein ist geeignet für einfache Dreipunkt-Schrittregelungen.

Die Steuerung des Stellgliedes erfolgt je nach Richtung der Abweichung Istwert-Sollwert in bezug auf den oberen und unteren Schwellwert. Die Schwellwertsteuerung beschreibt eine parametrierbare Hysterese.

Dieser Regler kann auch für Warm-/Kaltregelungen verwendet werden. Bei komplexeren Regelungen ist ein traditioneller Regler vorzuziehen (z.B. ein PI_B-Regler), dem ein Funktionsbaustein, wie beispielsweise der PWM1 nachzuschalten ist.

Als zusätzliche Parameter können EN und ENO projektiert werden.

Eigenschaften

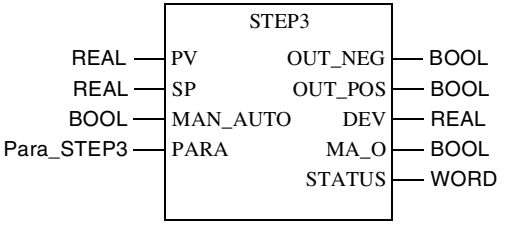
Der Reglerbaustein besitzt folgende Eigenschaften:

- Begrenzung des Sollwertes zwischen pv_inf und pv_sup
- Die Reglereingangswerte (Istwert, Sollwert und entsprechende Parameter) werden in physikalischen Einheiten ausgedrückt.

Darstellung

Symbol

Darstellung des Bausteins



Parameterbeschreibung STEP3

Beschreibung der Bausteinparameter

Parameter	Datentyp	Bedeutung
PV	REAL	Istwert (Process Value)
SP	REAL	Sollwert (Set Point)
MAN_AUTO	BOOL	Regler-Betriebsart: "1" : Automatik-Betrieb "0" : Halt-Betrieb
PARA	Para_STEP3	Parameter
OUT_NEG	BOOL	Logischer Ausgang: wird bei negativen Abweichungen auf 1 gesetzt
OUT_POS	BOOL	Logischer Ausgang: wird bei positiven Abweichungen auf 1 gesetzt
DEV	REAL	Abweichung (PV-SP)
MA_O	BOOL	Aktuelle Betriebsart des Funktionsbausteins (0: Hand, 1: Automatik)
STATUS	WORD	Statuswort

Parameterbeschreibung Para_STEP3

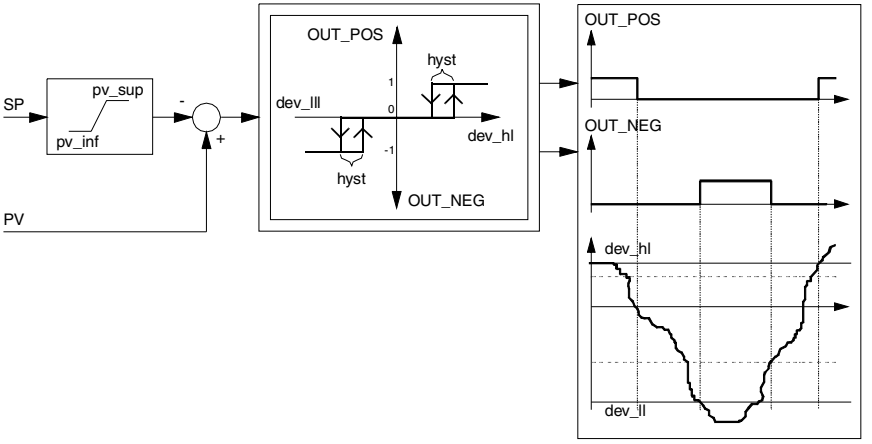
Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
dev_ll	REAL	Untere Schwelle der Abweichung (≤ 0)
dev_hl	REAL	Obere Schwelle der Abweichung (≥ 0)
hys	REAL	Hysterese
pv_inf	REAL	Untere Grenze des Istwertbereichs
pv_sup	REAL	Obere Grenze des Istwertbereichs

Detailbeschreibung

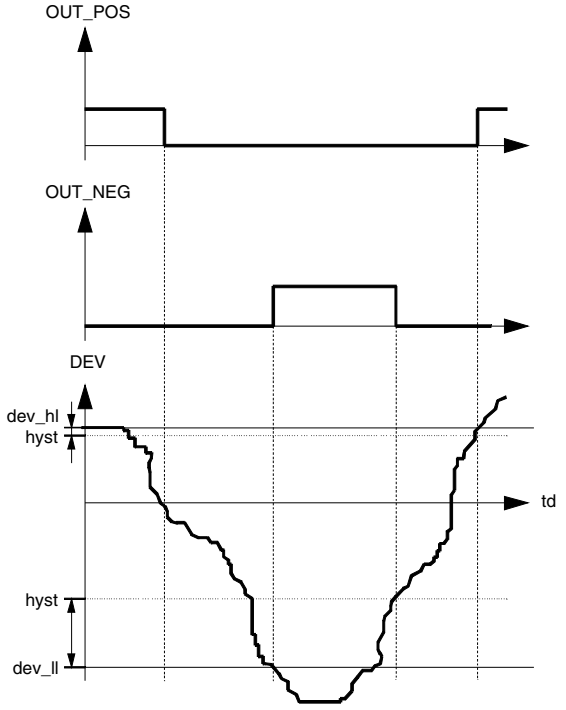
Strukturbild

Nachfolgend finden Sie das Strukturbild des Bausteins STEP3:



Verhalten der Ausgänge

Verhalten der Ausgänge OUT_POS und OUT_NEG:



td Zeitdauer

Wenn die Abweichung (DEV = PV - SP) über dev_hl hinaus ansteigt, wird der logische Ausgang OUT_POS auf 1 gesetzt. Wenn die Abweichung geringer wird, wird OUT_POS jedoch nur dann auf Null gesetzt, wenn die Abweichung dev_hl - hyst unterschreitet.

Wenn die Abweichung dev_ll unterschreitet, wird der logische Ausgang OUT_NEG auf 1 gesetzt. Wenn die Abweichung erneut ansteigt, wird OUT_NEG nur dann auf Null gesetzt, wenn die Abweichung über dev_ll + hyst hinaus ansteigt.

Hinweis: Um eine fehlerfreie Funktionsweise des Bausteins zu gewährleisten, dürfen die Ausgänge OUT_NEG und OUT_POS nicht invertiert werden.

Betriebsarten

Der Reglerbaustein STEP3 verfügt über 2 Betriebsarten, je nach Wert des Parameters MAN_AUTO :

Betriebsart	MAN_AUTO	Bedeutung
Automatik	1	Die Ausgänge OUT_NEG und OUT_POS werden vom Baustein selbst berechnet.
Halt	0	Die logischen Ausgänge werden vom Reglerbaustein nicht neu berechnet (Altwerte).

Laufzeitfehler

Statuswort

Im Statuswort werden folgende Meldungen angezeigt:

Bit	Bedeutung
Bit 0 = 1	Fehler in einer Berechnung mit Gleitkommawerten
Bit 1= 1	Erfassung eines unzulässigen Wertes an einem der Gleitkommawerteingänge
Bit 2= 1	Division durch Null bei einer Berechnung mit Gleitkommawerten
Bit 3 = 1	Kapazitätsüberlauf bei Berechnung in Gleitkommawerten
Bit 4 = 1	Folgende Verhalten werden angezeigt: • SP liegt außerhalb des Bereichs [pv_inf, pv_sup]: SP wird in diesem Fall auf pv_inf bzw. pv_sup begrenzt. • dev_ll > 0 oder dev_hl < 0: der Baustein verwendet den Wert 0 • hyst liegt außerhalb des Bereichs [0, Minimum (dev_hl, -dev_ll)]: der Baustein verwendet einen auf Null bzw. auf Minimum (dev_hl, -dev_ll) begrenzten Wert

Fehlermeldung

Ein Laufzeitfehler erscheint, wenn an einem Eingang ein nicht gleitender Wert erfaßt wird oder wenn bei einer Berechnung mit Gleitkommawerten ein Problem auftritt. Die Ausgänge OUT_NEG und OUT_POS werden in diesem Fall auf 0 gesetzt, die Ausgänge DEV und MA_O bleiben unverändert.

Warnung

In folgenden Fällen wird eine Warnung ausgegeben:

- dev_ll > 0 bzw. dev_hl < 0 : der Baustein verwendet den Wert 0.
- hyst liegt außerhalb des Bereichs [0, Minimum (dev_hl, -dev_ll)] : der Baustein verwendet einen begrenzten Wert.

SUM_W: Summierer

57

Übersicht

Einleitung

Dieses Kapitel beschreibt den Baustein SUM_W.

Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Kurzbeschreibung	514
Darstellung	514
Laufzeitfehler	514

Kurzbeschreibung

Funktionsbeschreibung

Der Funktionsbaustein führt die gewichtete Summation von 3 numerischen Eingangsgrößen nach untenstehender Formel durch.

Als zusätzliche Parameter können EN und ENO projektiert werden.

Formel

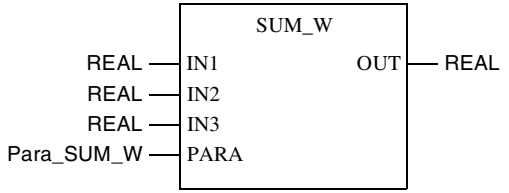
Der Baustein SUM_W arbeitet wie folgt:

$$OUT = k1 \times IN1 + k2 \times IN2 + k3 \times IN3 + c1$$

Darstellung

Symbol

Darstellung des Bausteins



Parameterbeschreibung SUM_W

Beschreibung der Bausteinparameter

Parameter	Datentyp	Bedeutung
IN1 bis IN3	REAL	Zu verarbeitende numerische Größen
PARA	Para_SUM_W	Parameter
OUT	REAL	Ergebnis der Berechnung

Parameterbeschreibung Para_SUM_W

Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
k1 bis k3, c1	REAL	Berechnungskoeffizienten

Laufzeitfehler

Fehlermeldung

Ein Laufzeitfehler erscheint, wenn an einem Eingang ein nicht gleitender Wert erfaßt wird oder wenn bei einer Berechnung mit Gleitkommawerten ein Problem auftritt. Der Ausgang OUT wird nicht verändert.

THREEPOINT_CON1: Dreipunkt-Regler

58

Übersicht

Einleitung

Dieses Kapitel beschreibt den Baustein THREEPOINT_CON1.

Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Kurzbeschreibung	516
Darstellung	516
Detailbeschreibung	518
Laufzeitfehler	521

Kurzbeschreibung

Funktionsbeschreibung

Der Funktionsbaustein bildet einen Dreipunktregler nach, der durch zwei dynamische Rückführungen PID-ähnliches Verhalten erhält.

Als zusätzliche Parameter können EN und ENO projektiert werden.

Eigenschaften

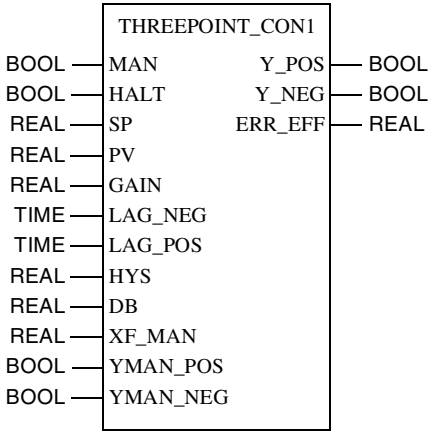
Der Funktionsbaustein THREEPOINT_CON1 besitzt folgende Eigenschaften:

- Betriebsarten Hand, Halt, Automatik
- zwei interne Rückführungen (Verzögerer 1. Ordnung)

Darstellung

Symbol

Darstellung des Bausteins



Parameterbeschreibung

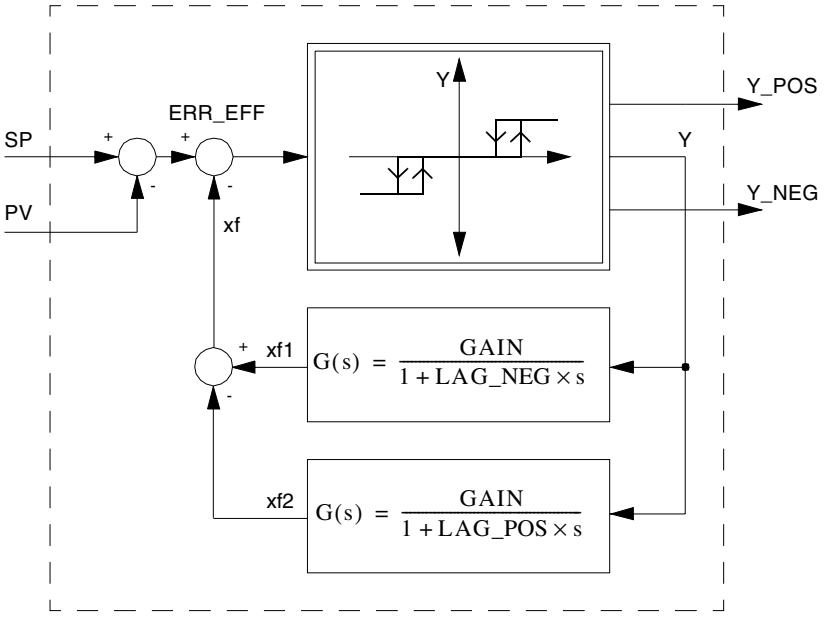
Beschreibung der Bausteinparameter

Parameter	Datentyp	Bedeutung
MAN	BOOL	"1" = Betriebsart Hand
HALT	BOOL	"1" = Betriebsart Halt
SP	REAL	Sollwerteingang
PV	REAL	Istwerteingang
GAIN	REAL	Rückführungsverstärkung (Rückführ-Parameter-Satz)
LAG_NEG	TIME	Zeitkonstante der schnellen Rückführung (Rückführ-Parameter-Satz)
LAG_POS	TIME	Zeitkonstante der langsamen Rückführung (Rückführ-Parameter-Satz)
HYS	REAL	Hysterese vom Dreipunktschalter
DB	REAL	Unempfindlichkeitszone
XF_MAN	REAL	Rücksetzwert der Rückführung in % (-100 bis 100)
YMAN_POS	BOOL	Hand-Stellwert für Y_POS
YMAN_NEG	BOOL	Hand-Stellwert für Y_NEG
Y_POS	BOOL	"1" = positive Stellgröße an Ausgang ERR_EFF
Y_NEG	BOOL	"1" = negative Stellgröße an Ausgang ERR_EFF
ERR_EFF	REAL	Wirksamer Schaltwert

Detailbeschreibung

Struktur des Reglers

Struktur des Dreipunktreglers:



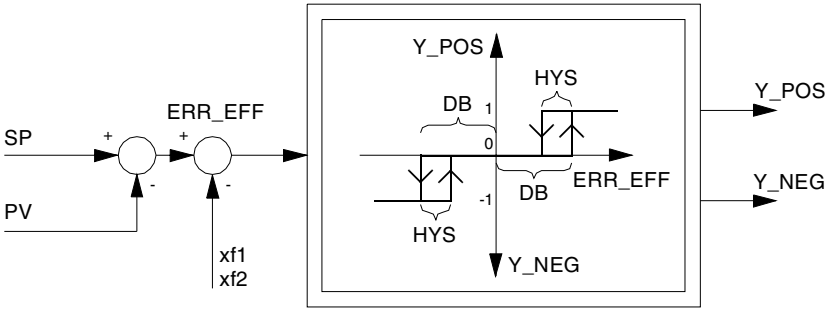
Abhängigkeit der Ausgänge Y_POS und Y_NEG von der Größe Y:

Wenn...	Dann...
Y = 1	Y_POS = 1 Y_NEG = 0
Y = 0	Y_POS = 0 Y_NEG = 0
Y = -1	Y_POS = 0 Y_NEG = 1

Prinzip des Dreipunktreglers

Dem eigentlichen Dreipunkt-Regler werden zwei dynamische Rückführungen (PT1-Glieder) hinzugefügt. Durch geeignete Wahl der Zeitkonstante dieser Rückführglieder erhält der Dreipunkt-Regler ein dynamisches Verhalten, das dem Verhalten eines PD-Reglers entspricht.

Prinzip des Dreipunktreglers



Der Parameter GAIN muß > 0 sein.

Hinweis: Für XF_MAN (Bedeutung -100 bis 100%) sind Werte zwischen -100 und 100 einzutragen!

Interne Rückführungen

Der Funktionsbaustein hat einen Parametersatz für die internen Rückführungen, bestehend aus der Rückführverstärkung GAIN und den Rückführzeitkonstanten LAG_NEG und LAG_POS.

Folgende Tabelle gibt dazu genaueren Aufschluß:

Rückführung	**LAG_NEG**	**LAG_POS**
3-Punkt-Verhalten (ohne Rückführung)	= 0	= 0
negative Rückführung	> 0	= 0
negative + positive Rückführung	> 0	> LAG_NEG
Warnung, da Mitkopplung (neg. Feedback mit LAG_POS)	= 0	> 0
Warnung, da Mitkopplung (pos. Feedback abgeschaltet)	> LAG_POS	> 0

Unempfindlichkeitszone

Der Parameter DB legt den Einschaltpunkt für die Ausgänge Y_POS und Y_NEG fest. Ist der wirksame Schaltwert ERR_EFF positiv und wird größer als DB, dann schaltet der Ausgang Y_POS von "0" auf "1". Ist wirksame Schaltwert ERR_EFF negativ und wird kleiner als -DB, dann schaltet der Ausgang Y_NEG von "0" auf "1". Der Wert des Parameters DB wird typisch auf 1% des maximalen Regelbereiches (max. SP - PV) eingestellt.

Hinweis: Von der Unempfindlichkeitszone DB wird der Betrag ausgewertet!

Hysterese

Der Parameter HYS gibt die Schalthysterese an, d.h. den Wert, um den sich der wirksame Schaltwert ERR_EFF ausgehend vom Schaltpunkt DB verkleinern muß bevor der Ausgang Y_POS (Y_NEG) wieder auf "0" gesetzt wird. Der Zusammenhang zwischen Y_POS und Y_NEG abhängig vom wirksamen Schaltwert ERR_EFF und den Parametern DB und HYS wird im Bild "*Prinzip des Dreipunktreglers, S. 519*" deutlich. Der Wert des Parameters HYS wird typisch auf 0.5% des maximalen Regelbereiches (max. SP - PV) eingestellt.

Hinweis: Von der Hysterese HYS wird der Betrag ausgewertet!

Betriebsarten

Es gibt drei Betriebsarten, die über die Eingänge MAN und HALT ausgewählt werden:

Betriebsart	MAN	HALT	Bedeutung
Automatik	0	0	Der Funktionsbaustein wird, wie vor beschrieben, bearbeitet.
Hand	1	0 oder 1	Die Ausgänge Y_POS und Y_NEG werden auf den Wert YMAN_POS und YMAN_NEG gesetzt. Hierbei ist die Vorranglogik - Y_NEG dominant über Y_POS - eingebaut, welche verhindert, daß beide Ausgänge gleichzeitig gesetzt sein können. xf1 und xf2 werden nach folgender Formel berechnet: $xf1 = XF_MAN \times \frac{GAIN}{100}$ $xf2 = XF_MAN \times \frac{GAIN}{100}$
Halt	0	1	Die beiden Ausgänge Y_POS und Y_NEG werden auf dem jeweils letzten Wert festgehalten. xf1 und xf2 werden auf GAIN * Y gesetzt.

Laufzeitfehler

Warnung

In folgenden Fällen erfolgt eine Warnung:

Wenn...	**Dann...**
LAG_NEG = 0 und LAG_POS > 0	arbeitet der Regler als hätte er nur eine negative Rückführung mit der Zeitkonstanten LAG_POS.
LAG_POS < LAG_NEG > 0	arbeitet der Regler als hätte er nur eine negative Rückführung mit der Zeitkonstanten LAG_NEG.
XF_MAN < -100 oder XF_MAN > 100	arbeitet der Regler ohne interne Rückführungen.

THREE_STEP_CON1: Dreipunkt-Schrittregler

59

Übersicht

Einleitung

Dieses Kapitel beschreibt den Baustein THREE_STEP_CON1.

Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Kurzbeschreibung	524
Darstellung	525
Detailbeschreibung	526
Laufzeitfehler	528

Kurzbeschreibung

Funktionsbeschreibung

Der Funktionsbaustein bildet einen Dreipunkt-Schrittregler nach, der durch eine dynamische Rückführung PD-ähnliches Verhalten erhält.

Als zusätzliche Parameter können EN und ENO projektiert werden.

Eigenschaften

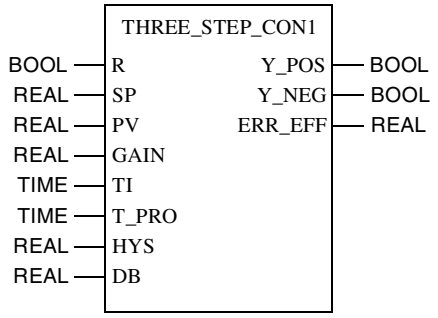
Der Funktionsbaustein THREE_STEP_CON1 besitzt folgende Eigenschaften:

- Betriebsarten Reset, Automatik
- eine interne Rückführung (Verzögerer 1. Ordnung)

Darstellung

Symbol

Darstellung des Bausteins



Parameterbeschreibung

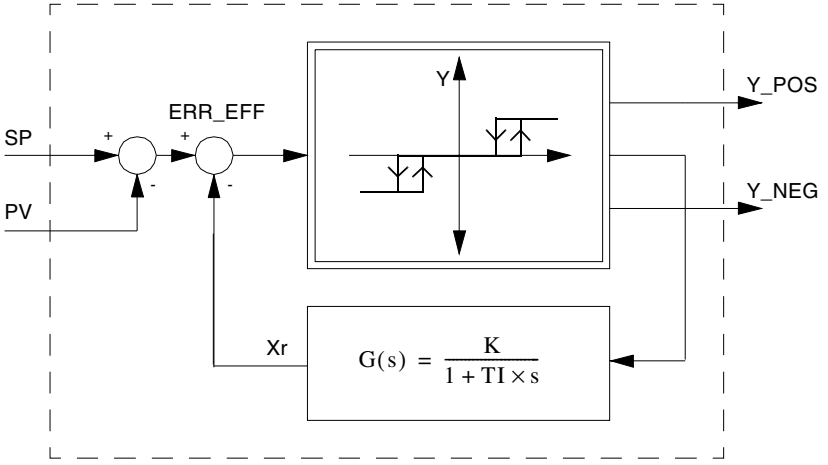
Beschreibung der Bausteinparameter

Parameter	Datentyp	Bedeutung
R	BOOL	"1" = Betriebsart Reset
SP	REAL	Sollwerteingang
PV	REAL	Istwerteingang
GAIN	REAL	Proportionalbeiwert (Verstärkungsfaktor)
TI	TIME	Nachstellzeit
T_PROC	TIME	Nennstellzeit des angeschlossenen Ventils
HYS	REAL	Hysterese vom Dreipunktschalter
DB	REAL	Unempfindlichkeitszone
ERR_EFF	REAL	Wirksamer Schaltwert
Y_POS	BOOL	"1" = positive Stellgröße an Ausgang ERR_EFF
Y_NEG	BOOL	"1" = negative Stellgröße an Ausgang ERR_EFF

Detailbeschreibung

Struktur des Reglers

Struktur des Dreipunktreglers:



Abhängigkeit der Ausgänge Y_POS und Y_NEG von der Größe Y:

Wenn...	Dann...
Y = 1	Y_POS = 1 Y_NEG = 0
Y = 0	Y_POS = 0 Y_NEG = 0
Y = -1	Y_POS = 0 Y_NEG = 1

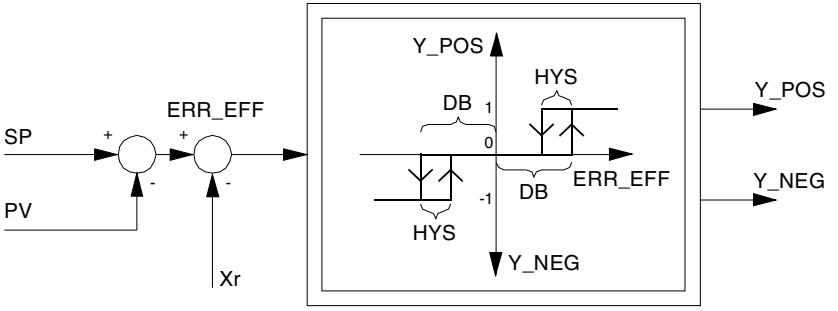
Bedeutung der Größe K:

$$K = \frac{TI}{T_PROC \times GAIN}$$

Prinzip des Dreipunktreglers

Dem eigentlichen Dreipunkt-Regler wird eine dynamische Rückführung (PT1-Glied) hinzugefügt. Durch geeignete Wahl der Zeitkonstante TI und T_PROC dieses Rückführgliedes erhält der Dreipunkt-Regler ein dynamisches Verhalten, das dem Verhalten eines PD-Reglers entspricht.

Prinzip des Dreipunktreglers



Der Parameter GAIN muß > 0 sein.

Unempfindlich-keitszone

Der Parameter DB legt den Einschaltpunkt für die Ausgänge Y_POS und Y_NEG fest. Ist der wirksame Schaltwert ERR_EFF = SP - PV- XR positiv und wird größer als DB, dann schaltet der Ausgang Y_POS von "0" auf "1". Ist der wirksame Schaltwert ERR_EFF negativ und wird kleiner als -DB, dann schaltet der Ausgang Y_NEG von "0" auf "1". Der Wert des Parameters DB wird typisch auf 1% des maximalen Regelbereiches (max. (SP - PV)) eingestellt.

Hinweis: Von der Unempfindlichkeitszone DB wird der Betrag ausgewertet!

Hysterese

Der Parameter HYS gibt die Schalthysterese an, d.h. den Wert, um den sich der wirksame Schaltwert ERR_EFF ausgehend vom Schaltpunkt DB verkleinern muß bevor der Ausgang Y_POS (Y_NEG) wieder auf "0" gesetzt wird. Der Zusammenhang zwischen Y_POS und Y_NEG abhängig von dem wirksamen Schaltwert ERR_EFF und den Parametern DB und HYS wird im Bild "*Prinzip des Dreipunktreglers, S. 527*" deutlich. Der Wert des Parameters HYS wird typisch auf 0.5 % des maximalen Regelbereiches (max. (SP - PV)) eingestellt.

Hinweis: Von der Hysterese HYS wird der Betrag ausgewertet!

Verhalten bei fehlerhafter Zeitkonstante

Ist die Zeitkonstante TI = 0 oder die Verstärkung GAIN ≤ 0 (Fehler in der Projektierung), so wird der Baustein dennoch bearbeitet. Allerdings wird die Funktion der Rückführung abgeschaltet, so daß der Baustein wie ein gewöhnlicher Dreipunktschalter arbeitet.

Ist die Zeitkonstante T_PROC = 0 (Fehler in der Projektierung), so wird der Baustein dennoch bearbeitet. In diesem Fall wird für T_PROC ein voreingestellter Wert von T_PROC = 60 s (60 000 msec) eingestellt.

Betriebsarten

Es gibt zwei Betriebsarten, die über den Eingang R ausgewählt werden:

Betriebsart	R	Bedeutung
Automatik	0	Der Funktionsbaustein wird, wie vor beschrieben, bearbeitet.
Reset	1	Der interne Wert des Rückführgliedes wird auf SP - PV gesetzt. Die Ausgänge Y_POS und Y_NEG werden beide auf "0" gesetzt.

Laufzeitfehler

Fehlermeldung

Bei HYS > 2 * DB erfolgt eine Fehlermeldung.

Warnung

In folgenden Fällen erfolgt eine Warnung:

Wenn...	Dann...
GAIN ≤ 0	arbeitet der Regler ohne Rückführung.
TI = 0	arbeitet der Regler ohne Rückführung.
T_PROC = 0	arbeitet der Regler mit einem voreingestellten Wert von T_PROC = 60 s

TOTALIZER: Integrierer

60

Übersicht

Einleitung

Dieses Kapitel beschreibt den Baustein TOTALIZER.

Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Kurzbeschreibung	530
Darstellung	531
Formeln	533
Detailbeschreibung	534
Laufzeitfehler	538

Kurzbeschreibung

Funktionsbeschreibung

Dieser Funktionsbaustein integriert den Wert des IN-Eingangs (typischerweise ein Durchflußvolumen) über die Zeit, bis ein regelbarer Grenzwert erreicht wird (typischerweise ein Volumen).

Als zusätzliche Parameter können EN und ENO projektiert werden.

Hinweis: Bei Verwendung des Freigabe-Eingangs EN ist folgendes zu beachten: Wenn der Baustein über einen längeren Zeitraum nicht aufgerufen wurde, weil der Freigabe-Eingang EN auf FALSE steht, verlängert sich die Laufzeit des Totalizer-Bausteins beim nächsten Aufruf. Bei einer Überschreitung des eingestellten Watchdog-Timeouts kann dies zu einem SPS-Stopp führen.
Zur Abhilfe sollte der Freigabe-Eingang nicht verwendet werden oder fest auf TRUE gesetzt werden, sodaß der Baustein in jedem Zyklus bearbeitet wird.

Eigenschaften

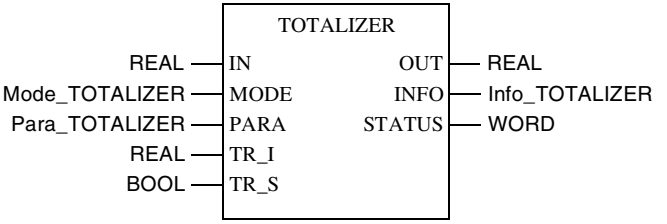
Der Funktionsbaustein besitzt folgende Eigenschaften

- Die Integration kann zeitweise angehalten und neu initialisiert werden
- Einrichtung, die auch sehr kleine Eingabewerte berücksichtigen kann
- Trennpunkt, bei dessen Unterschreitung die Werte von IN nicht mehr berücksichtigt werden
- Verwendung in der Betriebsart "Umkehrung der Integralsummation": der Ausgang OUT sinkt vom Schwellwert auf Null ab (inc_dec = 1)

Darstellung

Symbol

Darstellung des Bausteins



Parameterbeschreibung TOTALIZER

Beschreibung der Bausteinparameter

Parameter	Datentyp	Bedeutung
IN	REAL	Zu integrierende numerische Größe (nur wenn > 0)
MODE	Mode_TOTALIZER	Betriebsarten
PARA	Para_TOTALIZER	Parameter
TR_I	REAL	Initialisierungseingang von outc
TR_S	BOOL	Initialisierungsbefehl
OUT	REAL	Ergebnis der Integration von IN (auf thld begrenzt)
INFO	Info_TOTALIZER	vom Funktionsbaustein generierte Zusatzinformationen
STATUS	WORD	Statuswort

Parameterbeschreibung Mode_ TOTALIZER

Beschreibung der Datenstuktur

Element	Datentyp	Bedeutung
hold	BOOL	"1": Anhalten der Integration
rst	BOOL	"1": Reset des Funktionsbausteins

Parameterbeschreibung Para_ TOTALIZER

Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
thld	REAL	Integralschwelle von IN
cutoff	REAL	Trennpunkt (≥0)
inc_dec	BOOL	"1" : Umkehrung der Integration "0" : Normalbetrieb

Parameterbeschreibung Info_TOTALIZER

Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
outc	REAL	Gesamtergebnis der Integration von IN
cter	UINT	Zähler für Integralberechnung
done	BOOL	"1" : der Ausgang OUT erreicht den Integralschwellwert thld

Formeln

Berechnung des Ausgangs OUT

Bei jeder Ausführung wird der Ausgang OUT nach folgender Formel berechnet:

$$OUT(new) = OUT(old) + IN \times \Delta T$$

Wenn OUT den Schwellwert thld übersteigt:

- wird der Zähler cter inkrementiert: $cter = cter + 1$
- wird der Schwellwert thld vom Ausgang abgezogen: $OUT = OUT - thld$

Erläuterung der Formelgrößen

Bedeutung der Größen in den obigen Formeln:

Größe	Bedeutung
ΔT	seit der letzten Ausführung des Bausteins vergangene Zeit
$OUT(old)$	Wert des Ausgangs OUT bei Ende der vorherigen Ausführung des Reglers

Ausgabe der Integral-ergebnisse

Unter Berücksichtigung dieses Prinzips kann der Funktionsbaustein drei Integralergebnisse ausgeben:

Ergebnis	Erläuterung
Teilsammelregister OUT	gibt das Integralergebnis von Eingang IN ab der letzten Schwellwertüberschreitung an.
cter	Häufigkeit des Erreichens des Schwellwertes
Sammelregister (outc)	entspricht dem Integralergebnis des Eingangs IN seit Beginn der Integralrechnung. Dieser Zähler wird bei jeder Ausführung über folgende Formel aktualisiert: $outc = thld \times cter + OUT$

Detailbeschreibung

Einstellung der Integralschwelle thld

Der Integralschwellwert entspricht im allgemeinen einer Prozeßeigenschaft, die einfach zu bestimmen ist (z.B. der Inhalt eines Tanks).

Der Funktionsbaustein kann auch zur Integralberechnung kleiner Eingangswerte verwendet werden, auch dann, wenn das Ergebnis der Integralrechnung sehr groß ist. In diesem Fall besteht das Risiko, daß die Integralwerte sich im Verhältnis zum Gesamtwert zu stark abschwächen und daher nicht mehr berücksichtigt werden. Die vom TOTALIZER gebotene Lösung besteht in der Begrenzung des Sammelregisters OUT auf den Schwellwert thld, so daß der Integralwert im Verhältnis zum Teilsammelregister nie unbedeutend wird. Das Ergebnis des Gesamtintegrals (outc) wird ebenfalls berechnet: der Regler speichert die Häufigkeit des Erreichens des Schwellwertes thld am Sammelregister OUT.

Wenn der Schwellwert thld dem Wert 0 entspricht, wird der Integralwert nicht berechnet, die Ausgänge bleiben blockiert.

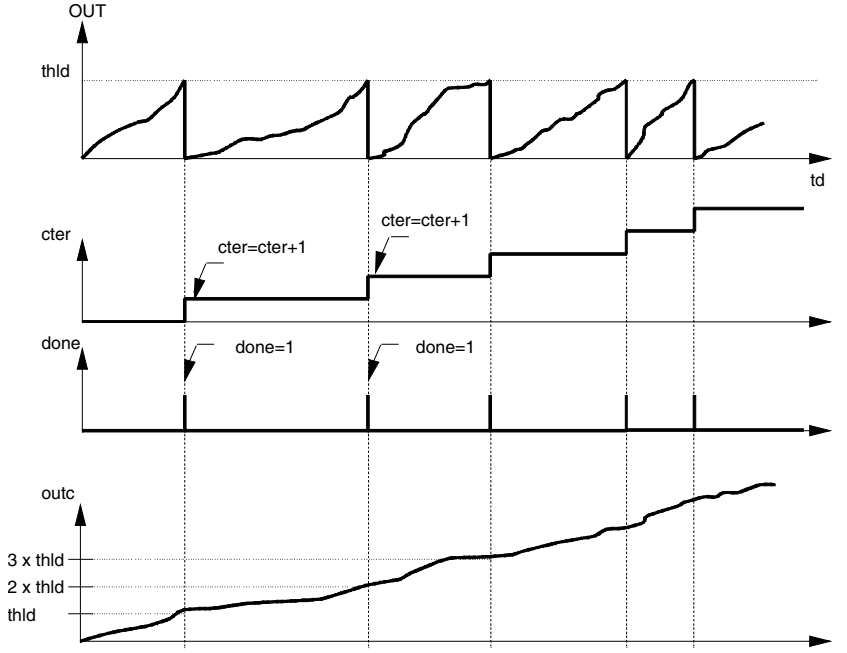
Weitere Eigenschaften

Sobald der Ausgang OUT den Schwellwert thld überschreitet, wird der Ausgang done auf 1 gesetzt. Bei der folgenden Ausführung des Funktionsbausteins wird dieser wieder auf Null gesetzt.

Wenn der Zähler cter seinen Maximalwert (65535) erreicht, ändert sich dieser Wert nicht mehr. Die Ausgänge OUT und done funktionieren weiterhin unter Einbeziehung des Schwellwertes thld, der Ausgang outc und der Zähler cter dürfen jedoch nicht mehr verwendet werden.

Die negativen Werte des Eingangs IN werden nie berücksichtigt, da diese immer unterhalb des Trennpunktes cutoff liegen.

Taktdiagramm Taktdiagramm des Bausteins TOTALIZER



td Zeitdauer

Betriebsarten

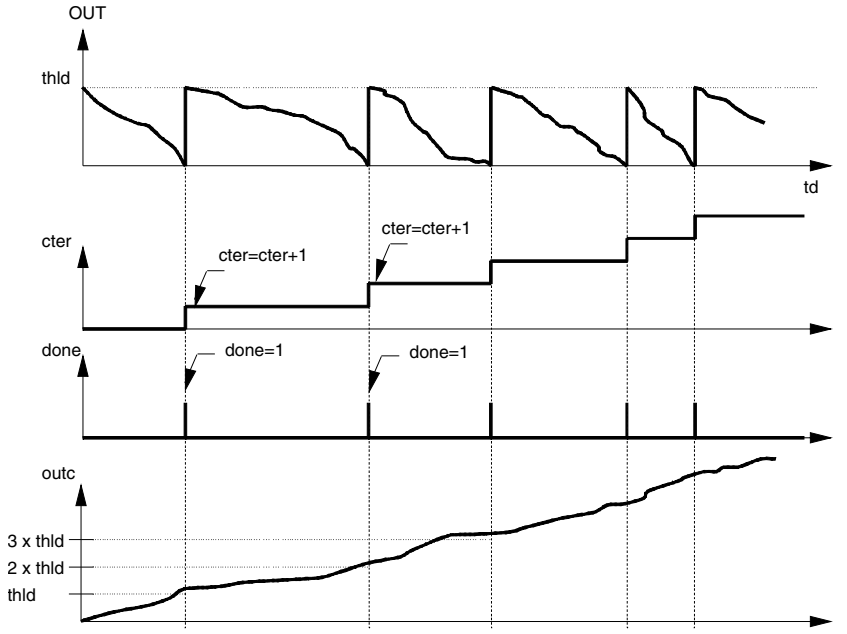
Der Funktionsbaustein TOTALIZER verfügt über drei individuelle Betriebsarten: Tracking, Reset und Halt:

Betriebsart	Parameter	Bedeutung
Tracking	TR_S = 1	Der Parameter TR_I wird auf outc geführt und die Parameter OUT und cter so gesetzt, daß folgende Gleichung gilt: outc= thld x cter + OUT. Die Betriebsart Tracking ermöglicht eine erneute Synchronisation der Reglerausgänge zum Regelprozeß (z.B. infolge eines Sensorausfalls).
Reset	rst = 1	Die Ausgänge OUT, outc, cter und done werden auf Null gesetzt. Der Reset über rst ermöglicht einen neuen Start ab dem Nullpunkt (beispielsweise nach Phasenänderung in der Produktion).
Halt	hold = 1	Die Integration wird angehalten. Die Ausgänge behalten ihre vorherigen Werte bei.

Hinweis: Bei gleichzeitiger Aktivierung der Eingänge TR_S, rst und hold hat der Trackingbetrieb Vorrang vor den anderen Betriebsarten und die Betriebsart Reset hat Vorrang vor Halt.

Umkehrung der Integral-summation (inc_dec = 1)

Darstellung des Funktionsprinzips



td Zeitdauer

Im Trackingbetrieb (TR_S = 1) wird der Parameter TR_I auf outc geführt und die Parameter OUT und cter werden so gesetzt, daß folgende Gleichung gilt:

outc = thld x cter + (thld -OUT).

outc wird nach folgender Formel berechnet: outc = thld x cter + (thld -OUT)

Funktions-prinzip der Umkehrung der Integral-summation

Es gilt folgendes Funktionsprinzip:

Step	Action
1	Bei der ersten Ausführung sowie bei positiver Flanke an rst wird der Ausgang OUT bei thld initialisiert.
2	Danach wird bei jeder Ausführung der Ausgang OUT nach folgender Formel berechnet: $OUT(new) = OUT(old) - IN \times \Delta T$
3	Sobald der Ausgang OUT negativ wird, geschieht folgendes: • Der Zähler cter wird inkrementiert: cter = cter + 1 • Der Schwellwert thld wird dem Ausgang OUT hinzuaddiert: OUT = OUT + thld • done wird auf 1 gesetzt

Laufzeitfehler

Statuswort

Im Statuswort werden folgende Meldungen angezeigt:

Bit	Bedeutung
Bit 0 = 1	Fehler in einer Berechnung mit Gleitkommawerten
Bit 1= 1	Erfassung eines unzulässigen Wertes an einem der Gleitkommawerteingängen
Bit 2= 1	Division durch Null bei einer Berechnung mit Gleitkommawerten
Bit 3 = 1	Kapazitätsüberlauf bei Berechnung in Gleitkommawerten
Bit 4 = 1	Der Eingang TR_I oder einer der Parameter thld bzw. cutoff sind negativ: zur Berechnung verwendet der Regler den Wert 0
Bit 6 = 1	Das Zählregister cter hat seinen Maximalwert erreicht (65535) : cter wird auf diesen Wert gesperrt und der Ausgang outc hat keine Bedeutung mehr. Die Ausgänge OUT und done hingegen können weiterhin verwendet werden.

Fehlermeldung

Ein Laufzeitfehler wird gemeldet, wenn an einem Eingang ein nicht gleitender Wert erfaßt wird oder wenn in einer Berechnung mit Gleitkommawerten ein Problem auftritt. Die Ausgänge OUT, outc, cter und done bleiben in diesem Fall unverändert.

Warnung

In folgenden Fällen wird eine Warnung ausgegeben:

Wenn...	Dann...
thld < 0	verwendet der Regler zur Berechnung den Wert 0
cutoff < 0	verwendet der Regler zur Berechnung den Wert 0
cter = 65535	bleibt cter auf diesen Wert blockiert und der Ausgang outc hat keine Bedeutung mehr. Die Ausgänge OUT und done können jedoch weiterhin verwendet werden.

TWOPOINT_CON1: Zweipunkt-Regler

61

Übersicht

Einleitung

Dieses Kapitel beschreibt den Baustein TWOPOINT_CON1.

Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Kurzbeschreibung	540
Darstellung	541
Detailbeschreibung	542
Laufzeitfehler	544

Kurzbeschreibung

Funktionsbeschreibung

Der Funktionsbaustein bildet einen Zweipunktregler nach, der durch zwei dynamische Rückführungen PID-ähnliches Verhalten erhält.

Als zusätzliche Parameter können EN und ENO projektiert werden.

Eigenschaften

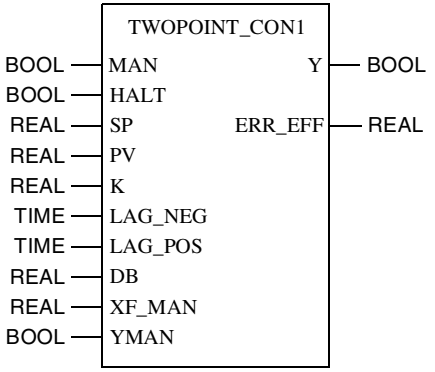
Der Funktionsbaustein TWOPOINT_CON1 besitzt folgende Eigenschaften:

- Betriebsarten Hand, Halt, Automatik
- Zwei interne Rückführungen (Verzögerer 1. Ordnung)

Darstellung

Symbol

Darstellung des Bausteins



Parameterbeschreibung

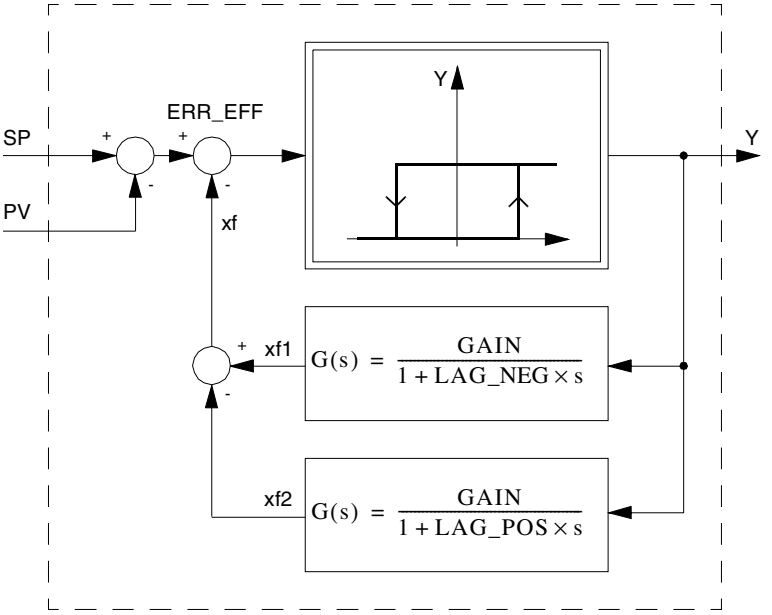
Beschreibung der Bausteinparameter

Parameter	Datentyp	Bedeutung
MAN	BOOL	"1" = Betriebsart Hand
HALT	BOOL	"1" = Betriebsart Halt
SP	REAL	Sollwerteingang
PV	REAL	Istwerteingang
K	REAL	Rückführverstärkung
LAG_NEG	TIME	Zeitkonstante der schnellen Rückführung
LAG_POS	TIME	Zeitkonstante der langsamen Rückführung
DB	REAL	Hysterese vom Zweipunktschalter
XF_MAN	REAL	Rücksetzwert der Rückführung in % (0 bis 100)
YMAN	BOOL	"1" = Handwert für ERR_EFF
Y	BOOL	"1" = Ausgang Stellgröße
ERR_EFF	REAL	Wirksamer Schaltwert

Detailbeschreibung

Struktur des Reglers

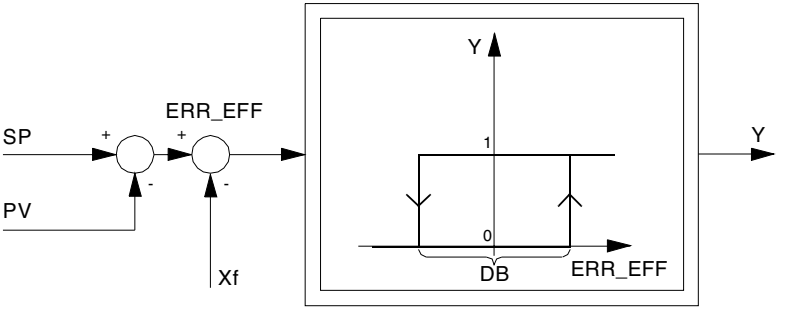
Struktur des Zweipunktreglers:



Prinzip des Zweipunkt-reglers

Dem eigentlichen Zweipunkt-Regler werden zwei dynamische Rückführungen (PT1-Glieder) hinzugefügt. Durch geeignete Wahl der Zeitkonstante dieser Rückführglieder erhält der Zweipunkt-Regler ein dynamisches Verhalten, das dem Verhalten eines PD-Reglers entspricht.

Prinzip des Zweipunktreglers:



Die Rückführverstärkung K ist größer Null zu wählen!

Für XF_MAN (Bedeutung 0 bis 100%) sind Werte zwischen 0 und 100 einzutragen!

Interne Rückführungen

Der Rückführparametersatz, bestehend aus der Rückführverstärkung K und den Rückführzeitkonstanten LAG_NEG und LAG_POS, ermöglicht einen universellen Einsatz des Zweipunktreglers.

Folgende Tabelle gibt dazu genaueren Aufschluß:

Rückführung	**LAG_NEG**	**LAG_POS**
2-Punkt-Verhalten (ohne Rückführung)	= 0	= 0
negative Rückführung	> 0	= 0
negative + positive Rückführung	> 0	> LAG_NEG
Warnung, da Mitkopplung (neg. Feadback mit LAG_POS)	= 0	> 0
Warnung, da Mitkopplung (pos. Feadback abgeschaltet)	> LAG_POS	> 0

Hysterese

Der Parameter DB gibt die Schalthysterese an, d.h. den Wert, um den sich der wirksame Schaltwert ERR_EFF ausgehend vom Schaltpunkt DB/2 verkleinern muß bevor der Ausgang Y wieder auf "0" gesetzt wird. Die Abhängigkeit des Ausgangs Y abhängig von dem wirksamen Schaltwert ERR_EFF und dem Parameter DB wird im Bild "*Prinzip des Zweipunktreglers, S. 543*" deutlich. Der Wert des Parameters DB wird typisch auf 1% des maximalen Regelbereich [max. (SP - PV)] eingestellt.

Betriebsarten

Es gibt drei Betriebsarten, die über die Eingänge MAN und HALT ausgewählt werden:

Betriebsart	MAN	HALT	Bedeutung
Automatik	0	0	Der Funktionsbaustein wird, wie vor beschrieben, bearbeitet.
Hand	1	0 oder 1	Der Ausgang Y wird auf den Wert YMAN gesetzt. xf1 und xf2 werden nach folgender Formel berechnet: $xf1 = XF_MAN \times \frac{GAIN}{100}$ $xf2 = XF_MAN \times \frac{GAIN}{100}$
Halt	0	1	Der Ausgang Y wird auf dem letzten Wert festgehalten. xf1 und xf2 werden auf GAIN * Y gesetzt.

Laufzeitfehler

Fehlermeldung

Bei HYS > 2 * DB erfolgt eine Fehlermeldung.

Warnung

In folgenden Fällen erfolgt eine Warnung:

Wenn...	Dann...
LAG_NEG = 0 und LAG_POS > 0	arbeitet der Regler als hätte er nur eine negative Rückführung mit der Zeitkonstanten LAG_POS.
LAG_POS < LAG_NEG > 0	arbeitet der Regler als hätte er nur eine negative Rückführung mit der Zeitkonstanten LAG_NEG.
XF_MAN < 0 oder XF_MAN > 100	arbeitet der Regler ohne interne Rückführungen.

VEL_LIM: Anstiegsbegrenzer

62

Übersicht

Einleitung

Dieses Kapitel beschreibt den Baustein VEL_LIM.

Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Kurzbeschreibung	546
Darstellung	547
Detailbeschreibung	548
Laufzeitfehler	549

Kurzbeschreibung

Funktionsbeschreibung

Der Funktionsbaustein realisiert einen Anstiegsbegrenzer mit Begrenzung der Stellgröße.

Es wird der Gradient der Eingangsgröße IN auf einen vorgebbaren Wert RATE begrenzt. Ferner wird der Ausgang OUT durch OUT_MAX und OUT_MIN begrenzt. Dadurch kann der Funktionsbaustein Signale an die technologisch begrenzte Geschwindigkeit und Endlagen von Stellgliedern anpassen.

Als zusätzliche Parameter können EN und ENO projektiert werden.

Eigenschaften

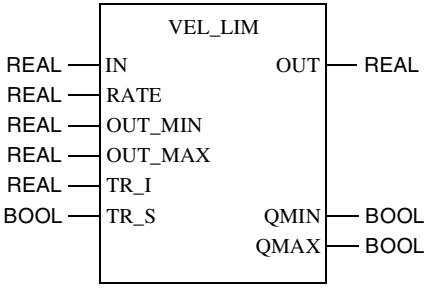
Der Funktionsbaustein besitzt folgende Eigenschaften:

- Betriebsarten Tracking und Automatik
- Stellgrößen-Begrenzung im Automatik-Betrieb

Darstellung

Symbol

Darstellung des Bausteins



Parameterbeschreibung

Beschreibung der Bausteinparameter

Parameter	Datentyp	Bedeutung
IN	REAL	Eingang
RATE	REAL	Maximale Anstiegsbegrenzung
OUT_MIN	REAL	Untere Stellgrenze
OUT_MAX	REAL	Obere Stellgrenze
TR_I	REAL	Initialisierungs-Eingang
TR_S	BOOL	Initialisierungs-Typ "1" = Betriebsart Tracking "0" = Betriebsart Automatik
OUT	REAL	Ausgang
QMIN	BOOL	"1" = Ausgang OUT, hat untere Grenze erreicht
QMAX	BOOL	"1" = Ausgang OUT, hat obere Grenze erreicht

Detailbeschreibung

Parametrierung

Die Parametrierung des Funktionsbausteins erfolgt durch Festlegen der maximalen Anstiegsgeschwindigkeit RATE sowie der Grenzen OUT_MAX und OUT_MIN für den Ausgang OUT. Die maximale Anstiegsgeschwindigkeit gibt an, um welchen Wert sich der Ausgang innerhalb einer Sekunde ändern darf.

Ist RATE = 0, wird OUT = IN.

Die Grenzen OUT_MAX und OUT_MIN begrenzen den Ausgang nach oben sowie nach unten. Damit gilt OUT_MIN ≤ OUT ≤ OUT_MAX.

Das Erreichen der Grenzwerte, bzw. eine Begrenzung des Ausgangssignals wird durch die beiden Ausgänge QMAX und QMIN angezeigt:

- QMAX = 1 wenn OUT ≥ OUT_MAX
- QMIN = 1 wenn OUT ≤ OUT_MIN

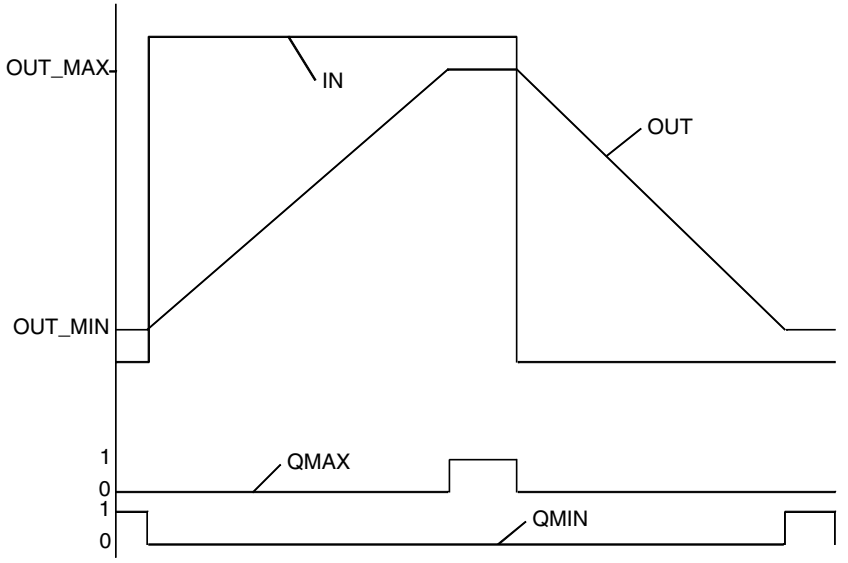
Betriebsarten

Es gibt zwei Betriebsarten, die über den Eingang TR_S ausgewählt werden:

Betriebsart	TR_S	Bedeutung
Automatik	0	Der aktuelle Wert für OUT wird ständig berechnet und ausgegeben.
Tracking	1	Der Trackingwert TR_I wird fest auf den Ausgang OUT übertragen. Der Stellausgang wird jedoch durch OUT_MAX und OUT_MIN begrenzt.

Beispiel

Darstellung des dynamischen Verhaltens des Funktionsbausteins VEL_LIM



Dem Sprung am Eingang IN folgt der Funktionsbaustein mit seiner maximalen Änderungsgeschwindigkeit. Deutlich zu erkennen ist auch die Begrenzung des Ausgangs OUT durch OUT_MAX und OUT_MIN mit den dazugehörigen Signalen QMAX und QMIN.

Laufzeitfehler

Fehlermeldung

Bei OUT_MAX < OUT_MIN erfolgt eine Fehlermeldung.

VLIM: Anstiegsbegrenzer 1. Ordnung

63

Übersicht

Einleitung

Dieses Kapitel beschreibt den Baustein VLIM.

Inhalt dieses Kapitels

Dieses Kapitel enthält die folgenden Themen:

Thema	Seite
Kurzbeschreibung	552
Darstellung	553
Detailbeschreibung	554
Laufzeitfehler	555

Kurzbeschreibung

Funktionsbeschreibung

Der Funktionsbaustein realisiert einen Anstiegsbegrenzer 1. Ordnung mit Begrenzung der Stellgröße.

Der Ausgang Y folgt dem Eingang X, jedoch mit dem maximalen Gradienten rate. Ferner wird der Ausgang Y durch ymax und ymin begrenzt. Dadurch kann der Funktionsbaustein Signale an die technologisch begrenzte Geschwindigkeit und Endlagen von Stellgliedern anpassen.

Als zusätzliche Parameter können EN und ENO projektiert werden.

Eigenschaften

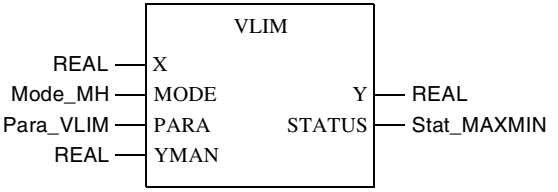
Der Funktionsbaustein besitzt folgende Eigenschaften:

- Betriebsarten Hand, Halt, Automatik
- Stellgrößen-Begrenzung

Darstellung

Symbol

Darstellung des Bausteins



Parameterbeschreibung VLIM

Beschreibung der Bausteinparameter

Parameter	Datentyp	Bedeutung
X	REAL	Eingang
MODE	Mode_MH	Betriebsarten
PARA	Para_VLIM	Parameter
YMAN	REAL	Hand-Stellwert
Y	REAL	Ausgang
STATUS	Stat_MAXMIN	Status des Ausgangs Y

Parameterbeschreibung Mode_VLIM

Beschreibung der Datenstuktur

Element	Datentyp	Bedeutung
man	BOOL	"1": Betriebsart Hand
halt	BOOL	"1": Betriebsart Halt

Parameterbeschreibung Para_VLIM

Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
rate	REAL	Maximale Anstiegsgeschwindigkeit (maximales x' / sec)
ymax	REAL	Obere Stellgrenze
ymin	REAL	Untere Stellgrenze

Parameterbeschreibung Stat_MAXMIN

Beschreibung der Datenstruktur

Element	Datentyp	Bedeutung
qmax	BOOL	"1" = Y hat obere Stellgrenze erreicht
qmin	BOOL	"1" = Y hat untere Stellgrenze erreicht

Detailbeschreibung

Parametrierung

Die Parametrierung des Funktionsbausteins erfolgt durch Festlegen der maximalen Anstiegsgeschwindigkeit rate sowie der Grenzen ymax und ymin für den Ausgang Y. Die maximale Anstiegsgeschwindigkeit gibt an, um welchen betragsmäßigen Wert sich der Ausgang innerhalb einer Sekunde maximal ändern darf.

Vom Parameter rate wird der Betrag ausgewertet.

Sonderfall rate = 0

Haben Sie rate=0 projektiert, so folgt der Ausgang Y dem Eingang X sofort (Y=X).

Stellgrenzen

Die Grenzen ymax und ymin begrenzen den Ausgang nach oben sowie nach unten. Damit gilt ymin ≤ Y ≤ ymax.

Das Erreichen der Grenzwerte, bzw. eine Begrenzung des Ausgangssignals wird durch die beiden Ausgänge qmax und qmin angezeigt:

- qmax = 1 wenn Y ≥ ymax
- qmin = 1 wenn Y ≤ ymin

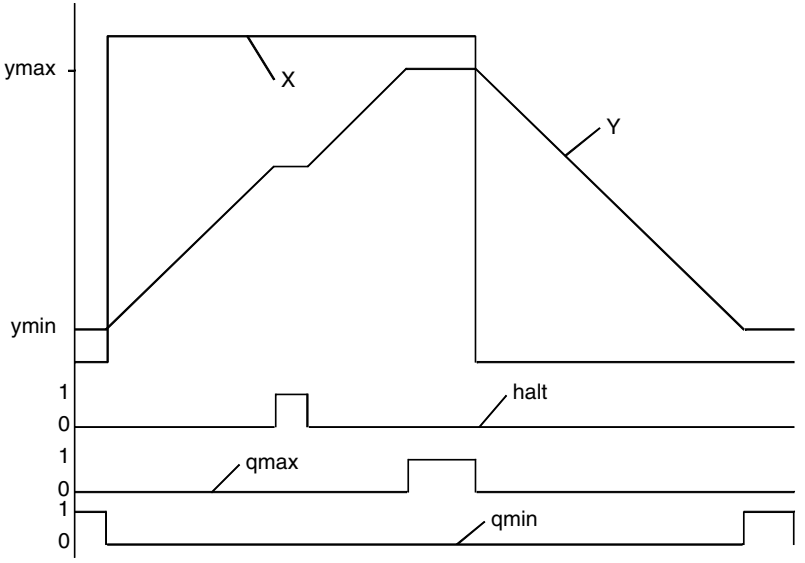
Betriebsarten

Es gibt drei Betriebsarten, die über die Eingänge MAN und HALT ausgewählt werden:

Betriebsart	MAN	HALT	Bedeutung
Automatik	0	0	Der aktuelle Wert für Y wird ständig berechnet und ausgegeben.
Hand	1	0 oder 1	Der Handwert YMAN wird fest auf den Ausgang Y übertragen. Der Stellausgang wird jedoch durch ymax und ymin begrenzt.
Halt	0	1	Der Ausgang Y wird auf dem letzten Wert gehalten. Der Ausgang wird nicht mehr verändert, kann jedoch vom Anwender überschrieben werden.

Beispiel

Darstellung des dynamischen Verhaltens des Funktionsbausteins VLIM



Dem Sprung am Eingang X folgt der Funktionsbaustein mit seiner maximalen Änderungsgeschwindigkeit (rate). Während der Betriebsart Halt bleibt der Ausgang Y stehen, um anschließend an der Stelle weiterzufahren, an der er stehen geblieben ist. Deutlich zu erkennen ist auch die Begrenzung des Ausgangs Y durch ymax und ymin mit den dazugehörigen Meldungen qmax und qmin.

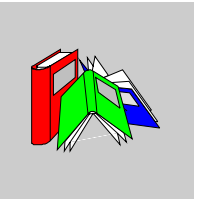
Laufzeitfehler

Fehlermeldung

Es erfolgt eine Fehlermeldung, wenn

- eine unzulässige Gleitkommazahl am Eingang YMAN oder X anliegt,
- ymax < ymin ist.

Glossar



A

Abgeleiteter Datentyp

Abgeleitete Datentypen sind Datentypen, die von den Elementaren Datentypen und/ oder anderen Abgeleiteten Datentypen abgeleitet wurden. Die Definition der Abgeleiteten Datentypen erfolgt im Datentyp-Editor von Concept.
Es wird zwischen globalen Datentypen und lokalen Datentypen unterschieden.

Ablaufsprache (SFC)

Die SFC-Sprachelemente ermöglichen das Unterteilen einer SPS-Programm-Organisationseinheit in eine Anzahl Schritte und Transitionen, die durch gerichtete Verbindungen untereinander verbunden sind. Zu jedem Schritt gehört eine Anzahl Aktionen, und mit jeder Transition ist eine Transitionsbedingung verbunden.

Adressen

(Direkte) Adressen sind Speicherbereiche auf der SPS. Diese befinden sich im Signalspeicher und können Ein-/Ausgangs-Baugruppen zugewiesen sein.
Die Anzeige/Eingabe von direkten Adressen ist in den folgenden Formaten möglich:

- Standard-Format (400001)
- Separator-Format (4:00001)
- Kompakt-Format (4:1)
- IEC-Format (QW1)

aktives Fenster

Das Fenster, das momentan angewählt ist. Zu einem gegebenen Zeitpunkt kann immer nur ein Fenster aktiv sein. Wenn ein Fenster aktiv wird, ändert sich die Farbe seiner Titelleiste, um es von den anderen Fenstern zu unterscheiden. Nicht angewählte Fenster sind inaktiv.

Aktualparameter

Aktuell angeschlossener Eingangs-/Ausgangs-Parameter.

ANL_IN ANL_IN steht für den Datentyp "Analog-Eingang" und wird für die Analogwertverarbeitung verwendet. Dem Datentyp werden die in der E/A-Bestückungsliste festgelegten 3x-Referenzen der konfigurierten Analog-Eingabe-Baugruppe automatisch zugeordnet und dürfen deshalb nur mit Unlocated Variablen belegt werden.

ANL_OUT ANL_OUT steht für den Datentyp "Analog-Ausgang" und wird für die Analogwertverarbeitung verwendet. Dem Datentyp werden die in der E/A-Bestückungsliste festgelegten 4x-Referenzen der konfigurierten Analog-Ausgabe-Baugruppe automatisch zugeordnet und dürfen deshalb nur mit Unlocated Variablen belegt werden.

Anweisung (IL) Anweisungen sind die "Befehle" der Programmiersprache IL. Jede Anweisung beginnt in einer neuen Zeile und wird von einem Operator ggf. mit Modifizierer und, falls erforderlich für die jeweilige Operation, von einem oder mehreren Operanen gefolgt. Falls mehrere Operanden verwendet werden, werden diese durch Kommas getrennt.Vor der Anweisung kann eine Marke stehen, die von einem Doppelpunkt gefolgt wird. Der Kommentar muß, falls vorhanden, das letzte Element der Zeile sein.

Anweisung (ST) Anweisungen sind die "Befehle" der Programmiersprache ST. Anweisungen müssen durch Semikolons abgeschlossen werden. Es können mehrere Anweisungen (durch Semikolons getrennt) in einer Zeile stehen.

Anweisungsliste (IL) IL ist eine Textsprache nach IEC 1131, in der Operationen, wie z. B. bedingte oder unbedingte Aufrufe von Funktionsbausteinen und Funktionen, bedingte oder unbedingte Sprünge usw. durch Anweisungen darsgestellt werden.

Anwendungsfenster Das Fenster, das den Arbeitsbereich, die Menüleiste und die Werkzeugleiste für das Anwendungsprogramm enthält. Der Name des Anwendungsprogramms erscheint in der Titelleiste. Ein Anwendungsfenster kann mehrere Dokumentfenster enthalten.In Concept entspricht das Anwendungsfenster einem Projekt.

ANY In der vorliegenden Version umfaßt "ANY" die Elementaren Datentypen BOOL, BYTE, DINT, INT, REAL, UDINT, UINT, TIME und WORD und davon abgeleitete Datentypen.

ANY_BIT In der vorliegenden Version umfaßt "ANY_BIT" die Datentypen BOOL, BYTE und WORD.

ANY_ELEM In der vorliegenden Version umfaßt "ANY_ELEM" die Datentypen BOOL, BYTE, DINT, INT, REAL, UDINT, UINT, TIME und WORD.

ANY_INT In der vorliegenden Version umfaßt "ANY_INT" die Datentypen DINT, INT, UDINT und UINT.

ANY_NUM In der vorliegenden Version umfaßt "ANY_NUM" die Datentypen DINT, INT, REAL, UDINT und UINT.

ANY_REAL In der vorliegenden Version umfaßt "ANY_REAL" den Datentyp REAL.

Argument Gleichbedeutend mit Aktualparameter.

ASCII-Modus American Standard Code for Information Interchange Der ASCII-Modus wird für die Kommunikation mit verschiedenen Host-Geräten eingesetzt. ASCII arbeitet mit 7 Datenbits.

Atrium Der PC based Controller befindet sich auf einer Standard AT-Platine, und kann innerhalb eines Host-Computers in einem ISA-Bussteckplatz betrieben werden. Die Baugruppe besitzt ein Motherboard (benötigt SA85-Treiber) mit zwei Steckplätzen für PC104-Daughter-Boards. Davon wird ein PC104-Daughter-Board als CPU und das andere zur INTERBUS-Steuerung eingesetzt.

Aufruf Der Vorgang, durch den die Ausführung einer Operation eingeleitet wird.

Ausdruck Ausdrücke bestehen aus Operatoren und Operanden.

Ausgangs-/ Merker-Bits (0x-Referenzen) Ein Ausgangs-/Merker-Bit kann dafür benutzt werden, reale Ausgangsdaten durch eine Ausgabeeinheit des Steuerungssystems zu steuern, oder einen oder mehrere diskrete Ausgänge im Signalspeicher zu definieren.Hinweis: Das x, das nach der ersten Ziffer des Referenztyps steht, repräsentiert einen fünfstelligen Speicherort im Anwender-Datenspeicher, z. B. hat die Referenz 000201 die Bedeutung eines Ausgangs- oder Merkerbits an Adresse 201 des Signalspeichers.

Ausgangs-/ Merkerworte (4x-Referenzen) Ein Ausgangs-/Merkerwort kann für die Speicherung numerischer Daten (binär oder dezimal) im Signalspeicher benutzt werden, oder auch für das Senden der Daten von der CPU an eine Ausgabeeinheit im Steuerungssystem.Hinweis: Das x, das nach der ersten Ziffer des Referenztyps steht, repräsentiert einen fünfstelligen Speicherort im Anwender-Datenspeicher, z. B. hat die Referenz 400201 die Bedeutung eines 16-Bit Ausgangs-/Merkerworts an Adresse 201 des Signalspeichers.

Ausgangsparameter (Ausgang) Ein Parameter, mit dem das/die Ergebnis(se) der Auswertung eines FFBs zurückgegeben wird/werden.

Auswertung Der Prozeß, durch den ein Wert für eine Funktion oder für die Ausgänge eines Funktionsbausteins während der Programmausführung ermittelt wird.

B

Backup-Datei (Concept-EFB)

Die Backup-Datei ist eine Kopie der letzten Quellcode-Datei. Der Name dieser Backup-Datei ist "backup??.c" (dabei wird angenommen, daß Sie von Ihrer Quellcode-Datei nie mehr als 100 Kopien haben). Die erste Backup-Datei hat den Namen "backup00.c".Falls Sie an der Definitions-Datei Änderungen vorgenommen haben, die beim EFB keine Schnittstellenveränderung hervorrufen, können Sie sich das Erzeugen einer Backup-Datei durch das Editieren Ihrer Quellcode-Datei ersparen (**Objekte** → **Quelle**).Wird eine Backup-Datei angelegt können Sie ihr den Namen Quell-Datei geben.

Basis-16-Literale

Basis-16-Literale dienen zur Angabe von ganzzahligen Werten im Hexadezimalsystem. Die Basis muß durch das Präfix 16# gekennzeichnet werden. Die Werte dürfen kein Vorzeichen (+/-) haben. Einzelne Unterstrichzeichen (_) zwischen den Ziffern sind nicht signifikant.

Beispiel
16#F_F oder 16#FF (dezimal 255)
16#E_0 oder 16#E0 (dezimal 224)

Basis-2-Literale

Basis-2-Literale dienen zur Angabe von ganzzahligen Werten im Dualsystem. Die Basis muß durch das Präfix 2# gekennzeichnet werden. Die Werte dürfen kein Vorzeichen (+/-) haben. Einzelne Unterstrichzeichen (_) zwischen den Ziffern sind nicht signifikant.

Beispiel
2#1111_1111 oder 2#11111111 (dezimal 255)
2#1110_0000 oder 2#11100000 (dezimal 224)

Basis-8-Literale

Basis-8-Literale dienen zur Angabe von ganzzahligen Werten im Oktalsystem. Die Basis muß durch das Präfix 8# gekennzeichnet werden. Die Werte dürfen kein Vorzeichen (+/-) haben. Einzelne Unterstrichzeichen (_) zwischen den Ziffern sind nicht signifikant.

Beispiel
8#3_77 oder 8#377 (dezimal 255)
8#34_0 oder 8#340 (dezimal 224)

Bibliothek Sammlung von Software-Objekten, die für die Wiederverwendung beim Programmieren neuer Projekte vorgesehen sind, oder sogar zum Bauen von neuen Bibliotheken. Beispiele sind die Bibliothek der Elementaren Funktionsbausteintypen.
EFB-Bibliotheken können in Gruppen unterteilt sein.

Binäre Verbindungen Verbindungen zwischen Ausgängen und Eingängen von FFBs vom Datentyp BOOL.

Bitfolge Ein Datenelement, das aus einem oder mehreren Bits besteht.

BOOL BOOL steht für den Datentyp "boolsch". Die Länge der Datenelemente ist 1 Bit (im Speicher abgelegt in 1 Byte). Der Wertebereich für Variablen dieses Datentyps ist 0 (FALSE) und 1 (TRUE).

Bridge Eine Bridge ist eine Einrichtung, die Netzwerke verbindet. Sie ermöglicht die Kommunikation zwischen Teilnehmern an den zwei Netzwerken. Jedes Netzwerk hat seine eigene Token-Rotationsfolge - das Token wird nicht durch Bridges weitergegeben.

BYTE BYTE steht für den Datentyp "Bit-Folge 8". Die Eingabe erfolgt als Basis-2-Literal, Basis-8-Literal oder Basis-16-Literal. Die Länge der Datenelemente ist 8 Bit. Ein numerischer Wertebereich kann diesem Datentyp nicht zugeordnet werden.

D

Datentypen Die Übersicht zeigt die Hierarchie der Datentypen, wie sie bei Ein- und Ausgängen von Funktionen und Funktionsbausteinen verwendet werden. Generische Datentypen sind durch die Vorsilbe "ANY" gekennzeichnet.

- ANY_ELEM
  - ANY_NUM
    ANY_REAL (REAL)
    ANY_INT (DINT, INT, UDINT, UINT)
  - ANY_BIT (BOOL, BYTE, WORD)
  - TIME
- System-Datentypen (IEC-Erweiterungen)
- Abgeleitet (von 'ANY'-Datentypen)

Daten-übertragungs-einstellungen Einstellungen, die festlegen, wie Informationen von Ihrem Programmiergerät zur SPS übertragen werden.

DCP-E/A-Station

Mit einem Distributed Control Prozessor (D908) können Sie ein dezentrales Netzwerk mit übergeordneter SPS einrichten. Beim Einsatz eines D908 mit dezentraler SPS betrachtet die übergeordnete SPS die dezentrale SPS wie eine dezentrale E/A-Station. Die D908 und die dezentrale SPS kommunizieren über den Systembus, wodurch sich eine hohe Leistung bei minimaler Auswirkung auf die Zykluszeit ergibt. Der Datenaustausch zwischen der D908 und der übergeordneten SPS findet mit 1.5 Megabit pro Sekunde über den dezentralen E/A-Bus statt. Eine übergeordnete SPS kann bis zu 31 (Adresse 2-32) D908-Prozessoren unterstützen.

DDE (Dynamic Data Exchange)

Die DDE-Schnittstelle ermöglicht einen dynamischen Datenaustausch zwischen zwei Programmen unter Windows. Der Anwender kann die DDE-Schnittstelle im erweiterten Monitor dazu verwenden, seine eigenen Anzeigeanwendungen aufzurufen. Mit dieser Schnittstelle kann der Anwender (d.h. der DDE Client) nicht nur Daten vom erweiterten Monitor (dem DDE Server) lesen, sondern auch Daten an die SPS über den Server schreiben. Der Anwender kann so Daten direkt in der SPS ändern, während er die Ergebnisse überwacht und analysiert.Bei der Verwendung dieser Schnittstelle kann der Anwender sein eigenes "Graphic-Tool", "Face Plate" oder "Tuning Tool" anlegen, und dieses in das System integrieren. Die Tools können in jeder beliebigen Sprache geschrieben werden, z.B. Visual Basic, Visual-C++, die DDE unterstützt. Die Tools werden aufgerufen, wenn der Anwender eine der Schaltflächen im Dialogfeld Erweiterter Monitor betätigt.Concept-Graphic-Tool: Durch die DDE-Verbindung zwischen Concept und Concept Graphic Tool können Signale einer Projektierung als Taktdiagramm dargestellt werden.

Definitions-Datei (Concept-EFB)

Die Definitions-Datei beinhaltet allgemeine Beschreibungs-Informationen zum gewählten EFB und seine Formalparameter.

Defragmentierung

Mit der Defragmentierung werden unbeabsichtigte Lücken (z.B. durch Löschen von unbenutzten Variablen entstanden) im Speicherbereich entfernt.

Deklaration

Mechanismus zur Festlegung der Definition eines Sprachelements. Eine Deklaration umfaßt normalerweise die Verbindung eines Bezeichners mit einem Sprachelement und die Zuordnung von Attributen wie Datentypen und Algorithmen.

Derived Function Block (DFB) (Abgeleiteter Funktionsbaustein)

Ein abgeleiteter Funktionsbaustein repräsentiert den Aufruf eines abgeleiteten Funktionsbausteintyps. Einzelheiten der grafischen Form des Aufrufs finden Sie in der Definition "Funktionsblock (Exemplar)". Im Gegensatz zu Aufrufen von EFB-Typen werden Aufrufe von DFB-Typen mit doppelten vertikalen Linien auf der linken und rechten Seite des rechteckigen Blocksymbols gekennzeichnet.
Der Rumpf eines abgeleiteten Funktionsbausteintyps wird in FBD-Sprache, LD-Sprache, ST-Sprache, IL-Sprache entworfen, allerdings nur in der aktuellen Version des Programmiersystems. Auch abgeleitete Funktionen können in der aktuellen Version noch nicht definiert werden.
Es wird zwischen lokalen und globalen DFBs unterschieden.

Dezentrales Netzwerk (DIO) Eine dezentrale Programmierung im Modbus Plus Netzwerk ermöglicht maximale Leistung beim Datentransfer und keine besonderen Anforderungen an Verknüpfungen. Die Programmierung eines dezentralen Netzes ist einfach. Für die Einrichtung des Netzes braucht keine zusätzliche Kontaktplanlogik erstellt zu werden. Durch entsprechende Einträge im Peer Cop Prozessor werden alle Anforderungen für die Datenübertragung erledigt.

DFB-Code Der DFB-Code ist der ausführbare DFB-Code einer Section. Die Größe des DFB-Codes ist hauptsächlich von der Bausteinanzahl in der Section abhängig.

DFB-Exemplardaten Die DFB-Exemplardaten sind interne Daten der im Programm verwendeten abgeleiteten Funktionsbausteine.

DINT DINT steht für den Datentyp "doppeltlange ganze Zahl (double integer)". Die Eingabe erfolgt als Integer-Literal, Basis-2-Literal, Basis-8-Literal oder Basis-16-Literal. Die Länge der Datenelemente ist 32 Bit. Der Wertebereich für Variablen dieses Datentyps reicht von -2 exp (31) bis 2 exp (31) -1.

Direkte Darstellung Eine Methode zur Variablendarstellung im SPS-Programm, aus der die Zuordnung zum logischen Speicherort - und indirekt zum physikalischen Speicherort - direkt abgeleitet werden kann.

Dokument-fenster Ein Fenster innerhalb eines Anwendungsfensters. Es können mehrere Dokument-fenster gleichzeitig in einem Anwendungsfenster geöffnet sein. Es kann aber immer nur ein Dokumentfenster aktiv sein.Dokumentfenster in Concept sind z. B. Sections, das Meldungs-Fenster, der Referenzdaten-Editor und die SPS-Konfiguration.

DP (PROFIBUS) DP = Dezentrale Peripherie

Dummy Eine Leerdatei, die aus einem Textkopf mit allgemeinen Dateiinformationen, wie z.B. Autor, Erstellungsdatum, EFB-Bezeichnung usw. besteht. Der Anwender muß die Fertigstellung dieser Dummy-Datei durch weitere Einträge vornehmen.

DX Zoom Diese Eigenschaft ermöglich Ihnen das Anschließen an ein Programmierobjekt, um dessen Datenwerte zu beobachten und zu ändern, falls notwendig.

E

E/A-Bestückungs-liste In der E/A-Bestückungsliste werden die E/A- und Experten-Baugruppen der verschiedenen Zentraleinheiten konfiguriert.

EFB-Code

Der EFB-Code ist der ausführbare Code aller verwendeten EFBs. Dazu zählen auch die verwendeten EFBs in DFBs.

Eingangsbits (1x-Referenzen)

Der 1/0-Zustand von Eingangsbits wird durch die Prozeßdaten gesteuert, die von einem Eingabegerät zur CPU gelangen.

Hinweis: Das x, das nach der ersten Ziffer des Referenztyps steht, repräsentiert einen fünfstelligen Speicherort im Anwender-Datenspeicher, z. B. hat die Referenz 100201 die Bedeutung eines Eingangsbits an Adresse 201 des Signalspeichers.

Eingangs-parameter (Eingang)

Übergibt beim Aufruf eines FFBs das zugehörige Argument.

Eingangsworte (3x-Referenzen)

Ein Eingangswort enthält Informationen, die von einer externen Quelle stammen und durch die eine 16-Bit Zahl repräsentiert wird. Ein 3x-Register kann auch 16 aufeinanderfolgende Eingangsbits beinhalten, die in das Register im Binär- oder BCD- (binary coded decimal) Format eingelesen wurden.Hinweis: Das x, das nach der ersten Ziffer des Referenztyps steht, repräsentiert einen fünfstelligen Speicherort im Anwender-Datenspeicher, z. B. hat die Referenz 300201 die Bedeutung eines 16-Bit-Eingangsworts an Adresse 201 des Signalspeichers.

Elementare Funktionen/ Funktionsbau-steine (EFB)

Bezeichnung für Funktionen oder Funktionsbausteine, deren Typdefinitionen nicht in einer der IEC-Sprachen formuliert sind, d.h. deren Rümpfe z. B. nicht mit dem DFB-Editor (Concept-DFB) modifiziert werden können. EFB-Typen werden in "C" programmiert und werden über Bibliotheken in vorkompilierter Form bereitgestellt.

EN / ENO (Freigabe / Fehleranzeige)

Falls der Wert von EN gleich "0" ist, wenn der FFB aufgerufen wird, werden die Algorithmen, die durch den FFB definiert sind, nicht ausgeführt und alle Ausgänge behalten ihren vorherigen Wert. Der Wert von ENO wird in diesem Fall automatisch auf "0" gesetztFalls der Wert von EN gleich "1" ist, wenn der FFB aufgerufen wird, werden die Algorithmen, die durch den FFB definiert sind, ausgeführt. Nach der fehlerfreien Ausführung dieser Algorithmen wird der Wert von ENO automatisch auf "1" gesetzt.Tritt ein Fehler während der Ausführung dieser Algorithmen auf, wird ENO automatisch auf "0" gesetzt.Das Ausgangsverhalten der FFBs ist unabhängig davon, ob die FFBs ohne EN/ENO oder mit EN=1 aufgerufen werden.Ist die Anzeige von EN/ENO eingeschaltet, muß der EN-Eingang unbedingt beschaltet werden. Andernfalls wird der FFB nie ausgeführt.Die Projektierung von EN und ENO wird im Dialogfeld der Bausteineigenschaften ein oder ausgeschaltet. Das Dialogfeld wird über die Menübefehle **Objekte** → **Eigenschaften...** oder über Doppelklick auf den FFB aufgerufen.

Exemplar-Name Ein Bezeichner, der zu einem bestimmten Funktionsbaustein-Exemplar gehört.Der Exemplar-Name dient der eindeutigen Kennzeichnung eines Funktionsbausteines in einer Programm-Organisationseinheit. Der Exemplar-Name wird automatisch erzeugt, kann aber editiert werden. Der Exemplar-Name muß in der gesamten Programm-Organisationseinheit eindeutig sein, dabei wird nicht zwischen Groß-/Kleinschreibung unterschieden. Existiert der eingegebene Name schon, werden Sie gewarnt und müssen einen anderen Namen wählen. Der Exemplar-Name muß den IEC-Namenskonventionen entsprechen, andernfalls erfolgt eine Fehlermeldung. Der automatisch erzeugte Instanzname hat immer den Aufbau: FBI_n_m

FBI = Funktionsbaustein-Exemplar
n = Nummer der Section (laufende Nummer)
m = Nummer des FFB-Objektes in der Section (laufende Nummer)

F

Fehler Wird bei der Abarbeitung eines FFBs oder eines Schrittes ein Fehler erkannt (z.B. unzulässige Eingangswerte oder ein Zeitfehler), erfolgt eine Fehlermeldung, die Sie sich mit dem Menübefehl **Online** → **Ereignisanzeige...** ansehen können. Bei FFBs wird der ENO-Ausgang auf "0" gesetzt.

Feldvariablen Variablen, denen ein mit Hilfe des Schlüsselworts ARRAY (Feld) ein definierter Abgeleiteter Datentyp zugeordnet ist.Ein Feld ist eine Sammlung von Datenelementen gleichen Datentyps.

FFB (Funktionen/ Funktionsbausteine) Sammelbegriff für EFB (Elementare Funktionen/Funktionsbausteine) und DFB (Abgeleitete (Derived) Funktionsbausteine)

FIR-Filter (Finite Impulse Response Filter) Filter mit endlicher Impulsantwort

Formalparameter Eingangs-/Ausgangsparameter, die innerhalb der Logik eines FFBs verwendet werden und als Ein-/Ausgänge aus dem FFB herausgeführt werden.

Funktion (FUNK)

Eine Programm-Organisationseinheit, die bei Ausführung genau ein Datenelement liefert. Eine Funktion hat keine interne Zustandsinformation. Mehrfachaufrufe derselben Funktion mit denselben Eingabeparameterwerten liefern immer dieselben Ausgabewerte.
Einzelheiten der grafischen Form von Funktionsaufrufen finden Sie in der Definition "Funktionsbaustein (Exemplar)". Im Gegensatz zu Aufrufen von Funktionsbausteinen haben Funktionsaufrufe nur einen einzelnen unbenannten Ausgang, da dessen Name der Name der Funktion selbst ist. In FBD wird jeder Aufruf durch eine eindeutige Nummer über dem grafischen Block gekennzeichnet; diese Nummer wird automatisch erzeugt und kann nicht verändert werden.

Funktionsbaustein (Exemplar) (FB)

Ein Funktionsbaustein ist eine Programm-Organisationseinheit, die entsprechend der in ihrer Funktionsbausteintypbeschreibung definierten Funktionalität Werte für ihre Ausgänge und interne(n) Variable(n) berechnet, wenn sie als bestimmtes Exemplar aufgerufen wird. Alle Werte der Ausgänge und internen Variablen eines bestimmten Funktionsbaustein-Exemplares bleiben von einem Aufruf des Funktionsbausteins bis zum nächsten bestehen. Mehrfachaufrufe desselben Funktionsbaustein-Exemplars mit denselben Argumenten (Werten von Eingangsparametern) liefern deshalb nicht unbedingt den-/dieselben Ausgabewert(e).
Jedes Funktionsbaustein-Exemplar ist grafisch durch ein rechteckiges Blocksymbol dargestellt. Der Name des Funktionsbausteintyps steht oben in der Mitte innerhalb des Rechtecks. Der Name des Funktionsbaustein-Exemplars steht auch oben, jedoch außerhalb des Rechtecks. Er wird automatisch bei Erstellung einer Instanz generiert, kann aber, je nach Bedarf, vom Anwender abgeändert werden. Eingänge werden auf der linken Seite, Ausgänge auf der rechten Seite des Blocks dargestellt. Die Namen der Formal-Ein-/Ausgangsparameter sind innerhalb des Rechtecks an den entsprechenden Stellen angezeigt.
Die obige Beschreibung der grafischen Darstellung gilt prinzipiell auch für Funktionsaufrufe und für DFB-Aufrufe. Unterschiede sind in den entsprechenden Definitionen beschrieben.

Funktionsbaustein-Sprache (FBD)

Eine oder mehrere Sections, die grafisch dargestellte Netzwerke aus Funktionen, Funktionsbausteinen und Verbindungen enthalten.

Funktionsbausteintyp

Ein Sprachelement, bestehend aus: 1. der Definition einer Datenstruktur, unterteilt in Eingangs-, Ausgangs-, und interne Variablen; 2. einem Satz von Operationen, die mit den Elementen der Datenstruktur durchgeführt werden, wenn eine Instanz des Funktionbausteintyps aufgerufen wird. Dieser Satz von Operationen kann entweder in einer der IEC-Sprachen (DFB-Typ) oder in "C" (EFB-Typ) formuliert sein.Ein Funktionsbausteintyp kann mehrfach instanziiert (aufgerufen) werden.

Funktionszähler Der Funktionszähler dient der eindeutigen Kennzeichnung einer Funktion in einem Programm oder DFB. Der Funktionszähler ist nicht editierbar und wird automatisch vergeben. Der Funktionszähler hat immer den Aufbau: .n.m

n = Nummer der Section (laufende Nummer)
m = Nummer des FFB-Objektes in der Section (laufende Nummer)

G

Generische Literale Wenn für Sie der Datentyp eines Literal nicht relevant ist, geben Sie einfach den Wert für das Literal an. In diesem Fall weist Concept dem Literal automatisch einen passenden Datentyp zu.

Generischer Datentyp Ein Datentyp, der stellvertretend für mehrere andere Datentypen steht.

Globale Abgeleitete Datentypen Globale Abgeleitete Datentypen sind in jedem Concept-Projekt verfügbar und sind im Verzeichnis DFB direkt unter dem Concept-Verzeichnis abgelegt.

Globale Daten Globale Daten sind Unlocated Variablen.

Globale DFBs Globale DFBs sind in jedem Concept-Projekt verfügbar. Die Ablage der globalen DFBs ist von der Einstellung in der CONCEPT.INI Datei abhängig.

Globale Makros Globale Makros sind in jedem Concept-Projekt verfügbar und sind im Verzeichnis DFB direkt unter dem Concept-Verzeichnis abgelegt.

Gruppen (EFBs) Manche EFB-Bibliotheken (z.B. die IEC-Bibliothek) sind in Gruppen unterteilt. Dies erleichtert besonders in umfangreichen Bibliotheken das Auffinden der EFBs.

H

Hochformat Hochformat bedeutet, daß die Seite bei Betrachtung des gedruckten Textes höher ist als breit.

I

IEC 61131-3
Internationale Norm: Speicherprogrammierbare Steuerungen - Teil 3: Programmiersprachen.

IEC-Format (QW1)
An erster Stelle der Adresse steht ein Bezeichner nach IEC, gefolgt von der fünfstelligen Adresse:
- %0x12345 = %Q12345
- %1x12345 = %I12345
- %3x12345 = %IW12345
- %4x12345 = %QW12345

IEC-Gesamtspeicher
Der IEC-Gesamtspeicher besteht aus dem IEC-Programmspeicher und den Globalen Daten.

IEC-Namens-konvention (Bezeichner)
Ein Bezeichner ist eine Folge von Buchstaben, Ziffern und Unterstrichzeichen, die mit einem Buchstaben oder Unterstrichzeichen beginnen muß (z.B. Name eines Funktionsbaustein-Typs, eines Exemplars, einer Variablen oder einer Section). Buchstaben aus Nationalen Zeichensätzen (z.B: ö,ü, é, õ) können verwendet werden, ausgenommen in Projekt- und DFB-Namen.
Unterstrichzeichen sind in Bezeichnern signifikant; z. B. "A_BCD" und "AB_CD" werden als unterschiedliche Bezeichner interpretiert. Mehrere führende und mehrfache Unterstrichzeichen hintereinander sind nicht zulässig.
Bezeichner dürfen keine Leerstellen enthalten.Groß- bzw. Kleinschreibung ist nicht signifikant; z. B. "ABCD" und "abcd" werden als der gleiche Bezeichner interpretiert. Bezeichner dürfen keine Schlüsselworte sein.

IEC-Programm-speicher
Der IEC-Programmspeicher besteht aus dem Programm-Code, EFB-Code, den Sectiondaten und den DFB-Exemplardaten.

IIR-Filter
(Infinite Impuls Response Filter) Filter mit unendlicher Impulsantwort

Initialschritt (Anfangsschritt)
Der Startschritt einer Ablaufkette. In jeder Ablaufkette muß ein Initialschritt definiert werden. Mit dem Initialschritt wird die Ablaufkette bei ihrem ersten Aufruf gestartet.

Initialwert
Der einer Variablen beim Programmstart zugeteilte Wert. Die Zuweisung des Wertes erfolgt in Form eines Literals.

Instanziierung
Die Erzeugung eines Exemplars.

Instruktion (LL984)

Beim Programmieren elektrischer Steuerungen hat ein Benutzer die Aufgabe, operational-codierte Anweisungen in Form von Bildobjekten, die in erkennbarer Kontaktform eingeteilt sind, zu implementieren. Die entworfenen Programmobjekte werden, auf Benutzerebene, während des Lade-Prozesses zu computer-verwendbaren OP-Codes konvertiert. Die OP-Codes werden in der CPU entschlüsselt und von den Firmware-Funktionen des Controllers so bearbeitet, daß die gewünschte Steuerung implementiert wird.

INT

INT steht für den Datentyp "ganze Zahl (integer)". Die Eingabe erfolgt als Integer-Literal, Basis-2-Literal, Basis-8-Literal oder Basis-16-Literal. Die Länge der Datenelemente ist 16 Bit. Der Wertebereich für Variablen dieses Datentyps reicht von -2 exp (15) bis 2 exp (15) -1.

Integer-Literale

Integer-Literale dienen der Angabe von ganzzahligen Werten im Dezimalsystem. Die Werte können ein vorangestelltes Vorzeichen (+/-) haben. Einzelne Unterstrich-zeichen (_) zwischen den Ziffern sind nicht signifikant.

Beispiel
-12, 0, 123_456, +986

INTERBUS (PCP)

Zur Nutzung des INTERBUS PCP-Kanals und der INTERBUS Prozeßdatenvorver-arbeitung (PDV) ist im Concept-Konfigurator der neue E/A-Stationstyp INTERBUS (PCP) eingeführt worden. Diesem E/A-Stationstyp ist die INTERBUS-Anschaltbau-gruppe 180-CRP-660-01 fest zugeordnet.
Die 180-CRP-660-01 unterscheidet sich von der 180-CRP-660-00 nur durch einen deutlich größeren E/A-Bereich im Signalspeicher der Steuerung.

K

Kompakt-Format (4:1)

Die erste Ziffer (die Referenz) wird durch einen Doppelpunkt (:) von der folgenden Adresse getrennt, wobei bei der Adresse die führenden Nullen nicht angegeben werden.

Konstanten

Konstanten sind Unlocated Variablen, denen ein Wert zugewiesen ist, der von der Programmlogik nicht geändert werden kann (schreibgeschützt).

Kontakt

Ein Kontakt ist ein LD-Element, das einen Zustand an die horizontale Verbindung auf seiner rechten Seite übergibt. Dieser Zustand ergibt sich aus der boolschen UND-Verknüpfung des Zustands der horizontalen Verbindung auf seiner linken Seite mit dem Zustand der zugehörigen Variablen/direkten Adresse. Ein Kontakt verändert nicht den Wert der zugehörigen Variablen/direkten Adresse.

L

Ladder Diagram (LD)

Ladder Diagram ist eine grafische Programmiersprache nach IEC1131, die sich optisch an den "Strompfaden" eines Relais-Kontaktplans orientiert.

Ladder Logic 984 (LL)

In den Begriffen Ladder Logic und Ladder Diagram bezieht sich das Wort Ladder (Kontakt) auf Ausführung. Im Gegensatz zu einem Schaltbild, wird ein Kontaktplan von Elektrotechnikern zum Zeichnen eines Stromkreises verwendet (anhand von elektrischen Symbolen), der den Ablauf von Ereignissen zeigen soll und nicht die vorhandenen Drähte, die die Teile miteinander verbinden. Eine gewöhnliche Benutzeroberfläche zum Steuern der Aktionen von Automatisierungsgeräten läßt eine Kontaktplanschnittstelle zu, damit Elektrotechniker zum Implementieren eines Steuerungsprogramms keine Programmiersprache lernen müssen, mit der sie nicht vertraut sind.
Der Aufbau des tatsächlichen Kontaktplanes ermöglicht das Verbinden der elektrischen Elemente auf eine Art und Weise, die einen Steuerungsausgang erzeugt, der von einem logischen Stromfluß durch die verwendeten elektrischen Objekte abhängig ist, die die vorher verlangte Bedingung eines physikalischen Elektrogerätes darstellt.
In einfacher Form ist die Benutzeroberfläche eine von der SPS-Programmieranwendung erarbeitete Video-Anzeige, die ein vertikales und horizontales Raster einrichtet, in das Programmierobjekte eingeordnet werden. Der Plan erhält an der linken Seite des Rasters Strom, und bei Verbindung mit Objekten, die aktiviert sind, fließt der Strom von links nach rechts.

Laufzeitfehler

Fehler, die während der Abarbeitung des Programms auf der SPS, bei SFC-Objekten (z. B. Schritten) oder FFBs auftreten. Dies sind z. B. Wertebereichs-Überläufe bei Zählern oder Zeitfehler bei Schritten.

Literale

Literale dienen dazu, Eingänge von FFBs, Transitionsbedingungen, usw. direkt mit Werten zu versorgen. Diese Werte können von der Programmlogik nicht überschrieben werden (schreibgeschützt). Dabei wird zwischen generischen und typisierte Litralen unterschieden.
Außerdem dienen Literale dazu, einer Konstanten einen Wert oder einer Variablen einen Initialwert zuzuweisen.
Die Eingabe erfolgt als Basis-2-Literal, Basis-8-Literal, Basis-16-Literal, Integer-Literal, Real-Literal oder Real-Literal mit Exponent.

Located Variable

Located Variablen wird eine Signalspeicheradresse (Referenzadressen 0x, 1x, 3x,4x) zugeordnet. Der Wert dieser Variablen wird im Signalspeicher gespeichert und kann mit dem Referenzdaten-Editor online geändert werden. Diese Variablen können mit ihrem symbloischen Namen oder mit ihrer Referenzadresse angesprochen werden.

Sämtliche Eingänge und Ausgänge der SPS sind mit dem Signalspeicher verbunden. Der Zugriff des Programms auf Peripherie-Signale, die an der SPS angeschlossen sind, erfolgt nur über Located Variablen. Zugriffe von externer Seite über Modbus- oder Modbus Plus-Schnittstellen der SPS, z.B. von Visualisierungas-systemen, sind ebenfalls über located Variablen möglich.

Lokale Abgeleitete Datentypen

Lokale Abgeleitete Datentypen sind nur in einem einzigen Concept-Projekt und dessen lokalen DFBs verfügbar und sind im Verzeichnis DFB unter dem Projekt-Verzeichnis abgelegt.

Lokale DFBs

Lokale DFBs sind nur in einem einzigen Concept-Projekt verfügbar und sind im Verzeichnis DFB unter dem Projekt-Verzeichnis abgelegt.

Lokale Makros

Lokale Makros sind nur in einem einzigen Concept-Projekt verfügbar und sind im Verzeichnis DFB unter dem Projekt-Verzeichnis abgelegt.

Lokale Verbindung (Local Link)

Die lokale Netzwerkverbindung ist das Netzwerk, das den lokalen Teilnehmer mit anderen Teilnehmern entweder direkt oder durch Busverstärker verbindet.

Lokaler Netzwerkteil-nehmer

Der lokale Teilnehmer ist der, der gerade projektiert wird.

M

Makro

Makros werden mit Hilfe der Software Concept-DFB erstellt.
Makros dienen zum Duplizieren häufig benutzter Sections und Netzwerke (inklusive deren Logik, deren Variablen und deren Variablendeklaration).
Es wird zwischen lokalen und globalen Makros unterschieden.

Makros haben die folgenden Eigenschaften:
- Makros können nur in den Programmiersprachen FBD und LD erstellt werden
- Makros enthalten nur eine einzige Section
- Makros können eine beliebig komplexe Section enthalten
- Programmtechnisch gesehen unterscheidet sich ein instanziertes Makro, d.h. ein in eine Section eingefügtes Makro, nicht von einer konventionell erstellen Section.
- Auruf von DFBs in einem Makro
- Deklaration von Variablen
- Verwendung von Makro-eigenen Datenstrukturen
- Automatische übernahme der im Makro deklarierten Variablen
- Initialwerte für Variablen
- Mehrfache Instanzierung eines Makros im Gesamtprogramm mit unterschiedlichen Variablen
- Der Sectionname, die Variablennamen und der Datenstrukturname können bis zu 10 verschiedene Austauschmarkierung (@0 bis @9) enthalten.

MMI

Mensch-Maschine-Interface

Multielement-Variablen

Variablen, denen ein mit STRUCT oder ARRAY definierter Abgeleiteter Datentyp zugeordnet ist.
Es wird dabei zwischen Feldvariablen und strukturierten Variablen unterschieden.

N

Netzwerk

Ein Netzwerk ist die Zusammenschaltung von Geräten an einem gemeinsamen Datenweg, die über ein gemeinsames Protokoll miteinander kommunizieren.

Netzwerkteil-nehmer

Ein Teilnehmer ist ein Gerät mit einer Adresse (1...64) am Modbus- Plus-Netzwerk.

Node Node ist eine Programmierzelle in einem LL984-Netzwerk. Eine Zelle/Node besteht aus einer 7x11 Matrix, d.h. 7 Reihen zu 11 Elementen.

O

Operand Ein Operand ist ein Literal, eine Variable, ein Funktionsaufruf oder ein Ausdruck.

Operator Ein Operator ist ein Symbol für eine auzuführende arithmetische oder boolsche Operation.

P

Peer-Prozessor Der Peer-Prozessor bearbeitet die Token-Durchläufe und den Datenfluß zwischen dem Modbus-Plus-Netzwerk und der SPS-Anwenderlogik.

Programm Die oberste Programm-Organisationseinheit. Ein Programm wird geschlossen auf eine einzelne SPS geladen.

Programm-Organisations-einheit Eine Funktion, ein Funktionsbaustein, oder ein Programm. Dieser Term kann sich entweder auf einen Typ oder ein Exemplar beziehen.

Programmier-gerät Hardware und Software, die das Programmieren, Projektieren, Testen, Inbetrieb-nehmen und die Fehlersuche in SPS-Anwendungen sowie in dezentralen Systemanwendungen unterstützt, um Quelldokumentation und -Archivierung zu ermöglichen. Das Programmiergerät kann u. U. auch für die Prozeßvisualisierung benutzt werden.

Programmzyklus Ein Programmzyklus besteht aus dem Einlesen der Eingänge, der Abarbeitung der Programmlogik und der Ausgabe der Ausgänge.

Projekt
Allgemeine Bezeichnung für die oberste Ebene einer Software-Baumstruktur, die den übergeordneten Projekt-Namen einer SPS-Anwendung festlegt. Nach der Festlegung des Projekt-Namens können Sie Ihre Systemkonfiguration und Ihr Steuerprogramm unter diesem Namen speichern. Alle Daten, die während der Erstellung der Konfiguration und des Programms entstehen, gehören zu diesem übergeordneten Projekt für diese spezielle Automatisierungs-Aufgabe.
Allgemeine Bezeichnung für den kompletten Satz von Programmier- und Projektierinformation in der Projektdatenbank, die den Quellcode darstellt, der die Automation einer Anlage beschreibt.

Projekt-datenbank
Die Datenbank im Programmiergerät, die die Projektierungs-Information für ein Projekt enthält.

Prototyp-Datei (Concept-EFB)
Die Prototyp-Datei beinhaltet alle Prototypen der zugeordneten Funktionen. Desweiteren wird, sofern vorhanden, eine Typdefinition der internen Zustandsstruktur angegeben.

Q

Quellcode-Datei (Concept-EFB)
Die Quellcode-Datei ist eine gewöhnliche C++ Quelldatei. Nach der Ausführung des Menübefehls **Bibliothek** → **Dateien erzeugen** enthält diese Datei einen EFB-Code-Rahmen, in den Sie einen spezifischen Code für den gewählten EFB eintragen müssen. Hierfür rufen Sie den Menübefehl **Objekte** → **Quelle** auf.

Querformat
Querformat bedeutet, daß die Seite bei Betrachtung des gedruckten Textes breiter ist als hoch.

R

REAL

REAL steht für den Datentyp "Gleitkomma-Zahl". Die Eingabe erfolgt als Real-Literal oder als Real-Literal mit Exponent. Die Länge der Datenelemente ist 32 Bit. Der Wertebereich für Variablen dieses Datentyps reicht von +/-3.402823E+38.

Hinweis: Abhängig vom Mathematikprozessortyp der CPU können verschiedene Bereiche innerhalb dieses zulässigen Wertebereichs nicht dargestellt werden. Dies gilt für Werte, die gegen NULL streben und für Werte, die gegen UNENDLICH streben. In diesen Fällen wird im Animationsmodus anstelle eines Zahlenwertes NAN (**N**ot **A** **N**umber) oder INF (**INF**inite (unendlich)) angezeigt.

Real-Literale

Real-Literale dienen zur Angabe von Gleitkomma-Werten im Dezimalsystem. Real-Literale werden durch die Angabe des Dezimalpunktes gekennzeichnet. Die Werte können ein vorangestelltes Vorzeichen (+/-) haben. Einzelne Unterstrichzeichen (_) zwischen den Ziffern sind nicht signifikant.

Beispiel
-12.0, 0.0, +0.456, 3.14159_26

Real-Literale mit Exponent

Real-Literale mit Exponent dienen zur Angabe von Gleitkomma-Werten im Dezimalsystem. Real-Literale mit Exponent werden durch die Angabe des Dezimalpunktes gekennzeichnet. Der Exponent gibt die Zehnerpotenz an, mit der die vorausgehende Zahl zu multiplizieren ist, um den darzustellenden Wert zu erhalten. Die Basis kann ein vorangestelltes negatives Vorzeichen (-) haben. Der Exponent kann ein vorangestelltes positives oder negatives Vorzeichen (+/-) haben. Einzelne Unterstrichzeichen (_) zwischen den Ziffern sind nicht signifikant. (Nur zwischen Ziffern, nicht vor oder hinter dem Dezimalpunkt und nicht vor oder hinter "E", "E+" oder "E-")

Beispiel
-1.34E-12 oder -1.34e-12
1.0E+6 oder 1.0e+6
1.234E6 oder 1.234e6

Redundanz-system programmieren (Hot Standby)

Ein Redundanzsystem besteht aus zwei identisch konfigurierten SPS-Geräten, die über Redundanzprozessoren miteinander kommunizieren. Im Falle eines Ausfalls der primären SPS übernimmt die sekundäre SPS die Steuerungskontrolle. Unter normalen Bedingungen übernimmt die sekundäre SPS keine Steuerungsfunktion, sondern überprüft die Statusinformationen, um Fehler zu erkennen.

Referenz

Jede direkte Adresse ist eine Referenz, die mit einer Kennung beginnt, die angibt, ob es sich um einen Eingang oder einen Ausgang und ob es sich um ein Bit oder ein Wort handelt. Referenzen, die mit der Kennzahl 6 beginnen, stellen Register im erweiterten Speicher des Signalspeichers dar.
0x Bereich = Ausgangs-/Merkerbits
1x Bereich = Eingangsbits
3x Bereich = Eingangsworte
4x Bereich = Ausgangs-/Merkerworte
6x Bereich = Register im erweiterten Speicher

Hinweis: Das x, das nach der ersten Ziffer jedes Referenztyps steht, repräsentiert einen fünfstelligen Speicherort im Anwender-Datenspeicher, z. B. hat die Referenz 400201 die Bedeutung eines 16-Bit Ausgangs- oder Merkerworts an Adresse 201 des Signalspeichers.

Register im erweiterten Speicher (6x-Referenz)

6x-Referenzen sind Merkerworte im erweiterten Speicher der SPS. Sie können nur bei LL984-Anwenderprogrammen und nur Verwendung einer CPU 213 04 oder CPU 424 02 verwendet werden.

RIO (Remote I/O)

Fern-E/A gibt einen physikalischen Ort der E/A-Punktsteuerungsgeräte an in Bezug auf den sie steuernden Prozessor. Fern-Ein-/Ausgänge sind mit dem Steuerungsgerät über ein verdrahtetes Kommunikationskabel verbunden.

RTU-Modus

Remote Terminal Unit
Der RTU-Modus wird für die Kommunikation zwischen der SPS und einem IBM-kompatiblen Personal Computer verwendet. RTU arbeitet mit 8 Datenbits.

S

SA85-Baugruppe

Die SA85-Baugruppe ist ein Modbus Plus-Adapter für IBM-AT oder kompatible Computer.

Schlüsselworte

Schlüsselworte sind eindeutige Zeichenkombinationen, die als spezielle syntaktische Elemente benutzt werden, wie dies in Anhang B der IEC 1131-3 definiert ist. Alle Schlüsselworte, die in der IEC 1131-3 und damit in Concept benutzt werden, sind im Anhang C der IEC 1131-3 aufgelistet. Diese aufgelisteten Schlüsselworte dürfen für keinen anderen Zweck verwendet werden, z. B. nicht als Variablen-Namen, Section-Namen, Exemplar-Namen, usw.

Schritt

SFC-Sprachelement: Situation, in der das Verhalten eines Programms in Bezug auf seine Eingänge und Ausgänge denjenigen Operationen folgt, die durch die zugehörigen Aktionen des Schrittes definiert sind.

Schrittname

Der Schrittname dient der eindeutigen Kennzeichnung eines Schrittes in einer Programm-Organisationseinheit. Der Schrittname wird automatisch erzeugt, kann aber editiert werden. Der Schrittname muß in der gesamten Programm-Organisationseinheit eindeutig sein, andernfalls erfolgt eine Fehlermeldung.
Der automatisch erzeugte Schrittname hat immer den Aufbau: S_n_m

S = Schritt
n = Nummer der Section (laufende Nummer)
m = Nummer des Schrittes in der Section (laufende Nummer)

Section

Eine Section kann z. B. dazu benutzt werden, die Funktionsweise einer technologischen Einheit, wie eines Motors, zu beschreiben.
Ein Programm oder DFB besteht aus einer oder mehreren Sections. Sections können mit den IEC-Programmiersprachen FBD und SFC programmiert werden. Innerhalb einer Section kann immer nur eine der genannten Programmiersprachen verwendet werden.
Jede Section hat in Concept ihr eigenes Dokumentfenster. Es ist aus Gründen der Übersichtlichkeit allerdings sinnvoll, eine sehr große Section in mehrere kleinere zu unterteilen. Zum Scrollen innerhalb der Section dient die Bildlaufleiste.

Section-Code

Section-Code ist der ausführbare Code einer Section. Die Größe des Section-Codes ist hauptsächlich von der Bausteinanzahl in der Section abhängig.

Sectiondaten

Sectiondaten sind die lokalen Daten in einer Section, wie z.B. Literals, Verbindungen zwischen Bausteinen, nicht verbundene Bausteinein- und ausgänge, interne Zustandsspeicher von EFBs.

Hinweis: Daten, die in DFBs dieser Section vorkommen, sind keine Sectiondaten.

Separator-(Trennzeichen)-Format (4:00001)

Die erste Ziffer (die Referenz) wird durch einen Doppelpunkt (:) von der folgenden fünfstelligen Adresse getrennt.

Serielle Anschlüsse

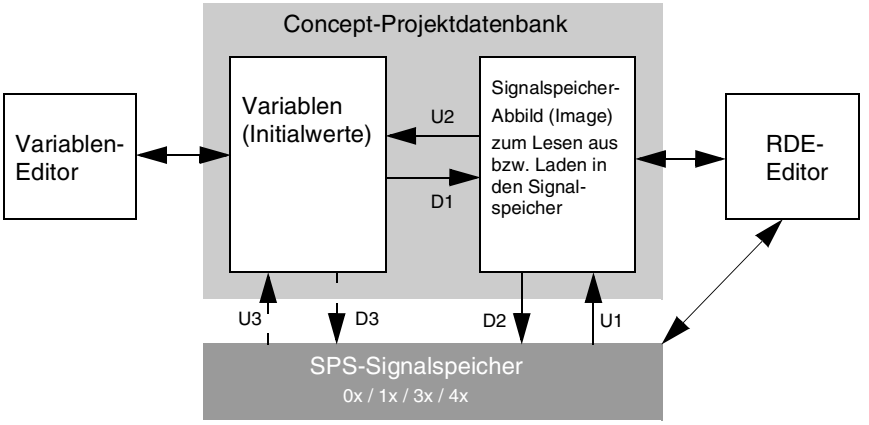
Bei seriellen Anschlüssen (COM) werden die Informationen bitweise übertragen.

Signalspeicher

Der Signalspeicher ist der Speicherplatz für alle Größen, die über Referenzen (Direkte Darstellung) im Anwenderprogramm angesprochen werden. Zum Beispiel stehen Eingangsbits, Ausgangs-/Merkerbits, Eingangsworte, und Ausgangs-/Merkerworte im Signalspeicher.

Signalspeicher-übersicht beim Auslesen und Laden

Übersicht:



Sprachelement Jedes Grundelement in einer der IEC-Programmiersprachen, z.B. ein Schritt in SFC, ein Funktionsbaustein-Exemplar in FBD oder der Anfangswert einer Variablen.

Sprung Element der SFC-Sprache. Sprünge werden verwendet, um Bereiche der Ablaufkette zu überspringen.

SPS Speicherprogrammierbare Steuerung

Spule Eine Spule ist ein LD-Element, das den Zustand der horizontalen Verbindung auf seiner linken Seite unverändert an die horizontale Verbindung auf seiner rechten Seite übergibt. Dabei wird der Zustand in der zugehörigen Variablen/direkten Adresse gespeichert.

Standard-Format (400001) Direkt nach der ersten Ziffer (der Referenz) steht die fünfstellige Adresse.

Status-Bits Zu jedem Teilnehmer mit globaler Eingabe oder spezifischer Ein-/Ausgabe von Peer Cop-Daten gibt es ein Status-Bit. Ist eine definierte Gruppe von Daten erfolgreich innerhalb des eingestellten Timeouts übertragen worden, so wird das entsprechende Status-Bit auf 1 gesetzt. Im anderen Fall wird dieses Bit auf 0 gesetzt und alle zu dieser Gruppe gehörenden Daten (zu 0) gelöscht.

Strukturierte Variablen Variablen, denen ein mit STRUCT (Struktur) definierter Abgeleiteter Datentyp zugeordnet ist.
Eine Struktur ist eine Sammlung von Datenelementen mit im allgemeinen unterschiedlichen Datentypen (Elementare Datentypen und/oder Abgeleitete Datentypen).

Strukturierter Text (ST) ST ist eine Textsprache nach IEC 1131, in der Operationen, wie z. B. Aufrufe von Funktionsbausteinen und Funktionen, bedingte Ausführung von Anweisungen, Wiederholung von Anweisungen usw. durch Anweisungen darsgestellt werden.

SY/MAX In Quantum-Steuergeräten schließt Concept die Bereitstellung auf E/A-Bestückung SY/MAX-E/A-Baugruppen für RIO-Steuerung durch die Quantum-SPS mit ein. Der SY/MAX-Fernbaugruppenträger hat einen Fern-E/A-Adapter in Steckplatz 1, der über ein Modicon S908 R E/A-System kommuniziert. Die SY/MAX-E/A-Baugruppen sind für Sie zur Markierung und Einbeziehung in die E/A-Bestückung der Concept-Konfiguration aufgeführt.

Symbol (Icon) Grafische Darstellung verschiedener Objekte in Windows, z. B. Laufwerke, Anwendungsprogramme und Dokumentfenster.

T

Teilnehmer-Adresse Die Teilnehmer-Adresse dient der eindeutigen Kennzeichnung eines Netzwerkteilnehmers im Routing-Pfad. Die Adresse wird direkt am Teilnehmer eingestellt, z.B. durch Drehschalter an der Rückseite der Baugruppen.

Template-Datei (Concept-EFB) Die Template-Datei ist eine ASCII-Datei mit einer Layout-Information für den Concept FBD-Editor, und den Parametern für die Code-Erzeugung.

TIME TIME steht für den Datentyp "Zeitdauer". Die Eingabe erfolgt als Zeitdauer-Literal. Die Länge der Datenelemente ist 32 Bit. Der Wertebereich für Variablen dieses Datentyps reicht von 0 bis 2exp(32)-1. Die Einheit für den Datentyp TIME ist 1 ms.

Token Das Netzwerk-"Token" steuert den vorübergehenden Besitz des Übertragungsrechtes durch einen einzelnen Teilnehmer. Das Token durchläuft die Teilnehmer in einer umlaufenden (aufsteigenden) Adressenfolge. Alle Teilnehmer verfolgen den Tokendurchlauf und können alle möglichen Daten erhalten, die mitgeschickt werden.

Traffic Cop Der Traffic Cop ist ist eine Bestückungsliste, die aus der Benutzer-Bestückungsliste generiert wird. Der Traffic Cop wird in der SPS verwaltet und enhält zusätzlich zur Anwender-Bestückungsliste z.B. Status-Informationen zu den E/A-Stationen und Baugruppen.

Transition Die Bedingung, bei der die Steuerung von einem oder mehreren Vorgängerschritten zu einem oder mehreren Nachfolgerschritten entlang einer gerichteten Verbindung übergeht.

Typisierte Literale

Wenn Sie den Datentyp für ein Literal selbst festlegen wollen, können Sie dies mit der folgenden Konstruktion tun: 'Datentypname'#'Wert des Literals'.

Beispiel
INT#15 (Datentyp: Integer, Wert: 15),
BYTE#00001111 (Datentyp: Byte, Wert: 00001111)
REAL#23.0 (Datentyp: Real, Wert: 23.0)

Für die Zuweisung des Datentyps REAL gibt es auch die Möglichkeit den Wert auf folgenden Weise anzugeben: 23.0.
Durch die Angabe der einer Kommastelle wird automatisch der Datentyp REAL zugewiesen.

U

UDEFB

Benutzer-definierte Elementare Funktionen/Funktionsbausteine
Funktionen oder Funktionsbausteine, die in der Programmiersprache C erstellt wurden, und die Concept in Bibliotheken zur Verfügung stellt.

UDINT

UDINT steht für den Datentyp "vorzeichenlose doppeltlange ganze Zahl (unsigned double integer)". Die Eingabe erfolgt als Integer-Literal, Basis-2-Literal, Basis-8-Literal oder Basis-16-Literal. Die Länge der Datenelemente ist 32 Bit. Der Wertebereich für Variablen dieses Datentyps reicht von 0 bis 2exp(32)-1.

UINT

UINT steht für den Datentyp "vorzeichenlose ganze Zahl (unsigned integer)". Die Eingabe erfolgt als Integer-Literal, Basis-2-Literal, Basis-8-Literal oder Basis-16-Literal. Die Länge der Datenelemente ist 16 Bit. Der Wertebereich für Variablen dieses Datentyps reicht von 0 bis (2exp 16)-1.

Unlocated Variable

Unlocated Variablen wird keine Signalspeicheradresse zugeordnet. Sie belegen somit auch keine Signalspeicheradressen. Der Wert dieser Variablen wird systemintern gespeichert und kann mit dem Referenzdaten-Editor gändert werden. Diese Variablen werden nur mit ihren symbolischen Namen angesprochen.

Signale die keinen Zugriff auf die Peripherie erfordern, z.B. Zwischenergebnisse, Systemmerker usw., sollten vorzugsweise als Unlocated Variable deklariert werden.

V

Variablen

Variablen dienen dem Datenaustausch innerhalb von Sections, zwischen mehreren Sections und zwischen dem Programm und der SPS.
Variablen bestehen mindestens aus einem Variablen-Namen und einem Datentyp. Wird einer Variablen eine direkte Adresse (Referenz) zugeordnet, spricht man von einer Located Variablen. Wird einer Variablen keine direkte Adresse zugeordnet, spricht man von einer Unlocated Variablen. Wird der Variablen ein Abgeleiteter Datentyp zugeordnet, spricht amn von einer Multielement-Variablen.
Außerdem gibt es noch Konstanten und Literale.

Verbindung

Eine Kontroll- oder Datenflußverbindung zwischen grafischen Objekten (z.B. Schritten im SFC-Editor, Funktionsbausteinen im FBD-Editor) innerhalb einer Section, grafisch als Linie dargestellt.

W

Warnung

Wird bei der Abarbeitung eines FFBs oder Schrittes ein kritischer Zustand erkannt (z.B. kritische Eingangswerte oder Zeitlimit überschritten), erfolgt eine Warnung, die Sie sich mit dem Menübefehl **Online** → **Ereignisanzeige...** ansehen können. Bei FFBs bleibt der ENO-Ausgang auf "1".

WORD

WORD steht für den Datentyp "Bit-Folge 16". Die Eingabe erfolgt als Basis-2-Literal, Basis-8-Literal oder Basis-16-Literal. Die Länge der Datenelemente ist 16 Bit. Ein numerischer Wertebereich kann diesem Datentyp nicht zugeordnet werden.

Z

Zeitdauer-Literale

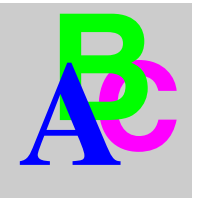
Erlaubte Einheiten für Zeitdauern (TIME) sind Tage (D), Stunden (H), Minuten (M), Sekunden (S) und Millisekunden (MS) oder Kombinationen hiervon. Die Zeitdauer muß durch das Präfix t#, T#, time# oder TIME# gekennzeichnet werden. Der "Überlauf" der höchstwertigen Einheit ist zulässig; z.B. ist die Eingabe T#25H15M erlaubt.

Beispiel
t#14MS, T#14.7S, time#18M, TIME#19.9H, t#20.4D, T#25H15M, time#5D14H12M18S3.5MS

Zwischenablage

Die Zwischenablage ist ein temporärer Speicher für ausgeschnittene oder kopierte Objekte. Diese Objekte können in Sections eingefügt werden. Bei jedem neuen Ausschneiden oder Kopieren wird der alte Inhalt der Zwischenablage überschrieben.

Index



A

ALIM, 41
Anstiegsbegrenzer, 545
Anstiegsbegrenzer 1. Ordnung, 207, 551
Anstiegsbegrenzer 2. Ordnung, 41
Automatische Reglereinstellung, 47
AUTOTUNE, 47

B

Baustein für Massendurchfluß, 213

C

CLC
- DELAY, 101
- INTEGRATOR1, 151
- LAG1, 167
- LEAD_LAG, 201
- LIMV, 207
- PI1, 281
- PID1, 319
- PIDP1, 379
- SMOOTH_RATE, 481
- THREE_STEP_CON1, 523
- THREEPOINT_CON1, 515
- TWOPOINT_CON1, 539

CLC_PRO
- ALIM, 41
- COMP_PID, 75
- DEADTIME, 95
- DERIV, 107
- FGEN, 123
- INTEG, 139
- LAG, 161
- LAG2, 173
- LEAD_LAG, 193
- PCON2, 235
- PCON3, 241
- PD_or_PI, 249
- PDM, 261
- PI, 271
- PID, 305
- PID_P, 331
- PID_PF, 341
- PIP, 389
- PPI, 401
- PWM, 411
- QPWM, 435
- SCON3, 459
- VLIM, 551

COMP_DB, 71
COMP_PID, 75

Conditioning
DTIME, 113
INTEGRATOR, 145
LAG_FILTER, 179
LDLG, 183
LEAD, 189
MFLOW, 213
QDTIME, 429
SCALING, 455
TOTALIZER, 529
VEL_LIM, 545
CONT_CTL
ALIM, 41
AUTOTUNE, 47
COMP_DB, 71
COMP_PID, 75
DEADTIME, 95
DELAY, 101
DERIV, 107
DTIME, 113
Einleitung, 27
FGEN, 123
INTEG, 139
INTEGRATOR, 145
INTEGRATOR1, 151
K_SQRT, 157
LAG, 161
LAG_FILTER, 179
LAG1, 167
LAG2, 173
LDLG, 183
LEAD, 189
LEAD_LAG, 193
LEAD_LAG1, 201
LIMV, 207
MFLOW, 213
MS, 219
MULDIV_W, 231
PCON2, 235
PCON3, 241
PD_or_PI, 249
PDM, 261
PI, 271
PI_B, 291
PI1, 281
PID, 305
PID_P, 331
PID_PF, 341
PID1, 319
PIDFF, 351
PIDP1, 379
PIP, 389
PPI, 401
PWM, 411
PWM1, 421
QDTIME, 429
QPWM, 435
RAMP, 443
RATIO, 449
SCALING, 455
SCON3, 459
SERVO, 465
SMOOTH_RATE, 481
SP_SEL, 487
SPLRG, 495
STEP2, 501
STEP3, 507
SUM_W, 513
THREE_STEP_CON1, 523
THREEPOINT_CON1, 515
TOTALIZER, 529
TWOPOINT_CON1, 539
VEL_LIM, 545
VLIM, 551
Controller
AUTOTUNE, 47
PI_B, 291
PIDFF, 351
STEP2, 501
STEP3, 507

D

DEADTIME, 95
DELAY, 101
DERIV, 107
Differenzierer mit Glättung, 107, 189, 481
Dreipunkt-Regler, 241, 515
Dreipunktregler, 507
Dreipunkt-Schrittregler, 459, 523
DTIME, 113

E

Einfacher PI-Regler, 291
Einleitung zur Baustein-Bibliothek
CONT_CTL, 27

F

FGEN, 123
Funktion
    Parametrierung, 23, 24
Funktionsbaustein
    Parametrierung, 23, 24
Funktionsgenerator, 123

H

Handsteuerung eines Ausgangs, 219

I

INTEG, 139
INTEGRATOR, 145
INTEGRATOR1, 151
Integrierer, 529
Integrierer mit Begrenzung, 139, 145, 151

K

K_SQRT, 157
Kompletter PID-Regler, 351
Komplexer PID-Regler, 75

L

LAG, 161
LAG_FILTER, 179
LAG1, 167
LAG2, 173
LDLG, 183
LEAD, 189
LEAD_LAG, 193
LEAD_LAG1, 201
LIMV, 207

M

Mathematics
    COMP_DB, 71
    K_SQRT, 157
    MULDIV_W, 231
    SUM_W, 513
MFLOW, 213
MS, 219
MULDIV_W, 231
Multiplikation/Division, 231

O

Output processing
    MS, 219
    PWM1, 421
    SERVO, 465
    SPLRG, 495

P

Parametrierung, 23, 24
PCON2, 235
PCON3, 241
PD_or_PI, 249
PD-Glied mit Glättung, 183, 193, 201
PDM, 261
PI, 271
PI_B, 291
PI1, 281
PID, 305
PID_P, 331
PID_PF, 341
PID1, 319
PIDFF, 351
PIDP1, 379
PID-Regler, 305, 319
PID-Regler mit paralleler Struktur, 331, 341, 379
PIP, 389
PIP-Kaskaden-Regler, 389
PI-Regler, 271, 281
PPI, 401
PPI-Kaskaden-Regler, 401
Pulsbreitenmodulation, 411, 421

Pulsbreitenmodulation (einfach), 435
Pulsdauermodulation, 261
PWM, 411
PWM1, 421

Q

QDTIME, 429
QPWM, 435
Quadratwurzel, 157

R

RAMP, 443
Rampengenerator, 443
RATIO, 449

S

SCALING, 455
SCON3, 459
SERVO, 465
Setpoint management
    RAMP, 443
    RATIO, 449
    SP_SEL, 487
Skalierung, 455
SMOOTH_RATE, 481
Sollwertschalter, 487
SP_SEL, 487
SPLRG, 495
STEP2, 501
STEP3, 507
Steuerung für elektrische Servomotoren, 465
Steuerung von 2 Stellgliedern, 495
Strukturumschaltung PD- /PI-Regler, 249
SUM_W, 513
Summierer, 513

T

THREE_STEP_CON1, 523
THREEPOINT_CON1, 515
TOTALIZER, 529
Totzeitglied, 95, 101, 429
TWOPOINT_CON1, 539

V

VEL_LIM, 545
Vergleich, 71
Verhältnisregler, 449
Verzögerung, 113
Verzögerungsglied 1. Ordnung, 161, 167, 179
Verzögerungsglied 2. Ordnung, 173
VLIM, 551

Z

Zweipunkt-Regler, 235, 539
Zweipunktregler, 501